Joint Transportation Research Program

JTRP

FHWA/IN/JTRP-98/7 -01

Final Report

A TOOL FOR EVALUATING ACCESS CONTROL ON HIGH-SPEED URBAN ARTERIALS

PART I: RESEARCH REPORT

Henry C. Brown
Samuel Labi
Andrzej P. Tarko
Jon D. Fricker

September 1998

Indiana Department of Transportation

Purdue University

Final Report

FHWA/IN/JTRP-98/7

A TOOL FOR EVALUATING ACCESS CONTROL ON HIGH-SPEED URBAN ARTERIALS PART I: RESEARCH REPORT

Henry C. Brown
Graduate Research Assistant
and
Samuel Labi
Graduate Research Assistant

and

Andrzej P. Tarko
Assistant Professor of Civil Engineering
and
Jon D. Fricker
Professor of Civil Engineering

School of Civil Engineering
Purdue University

Joint Transportation Research Program
Project No.: C-36-17UU
File No.: 8-4-47

Prepared in Cooperation with the
Indiana Department of Transportation and
the U.S. Department of Transportation
Federal Highway Administration

Purdue University
West Lafayette, Indiana,
September 1998

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA/IN/JTRP-98/7		

4. Title and Subtitle A Tool for Evaluating Access Control on High-Speed Urban Arterials (2 Volumes) Part I: Research Report Part II: User's Guide	5. Report Date September 1998
	6. Performing Organization Code
7. Author(s) Henry C. Brown, Samuel Labi, Andrzej P. Tarko, and Jon D. Fricker	8. Performing Organization Report No. FHWA/IN/JTRP-98/7
9. Performing Organization Name and Address Joint Transportation Research Program 1284 Civil Engineering Building Purdue University West Lafayette, Indiana 47907-1284	10. Work Unit No.
	11. Contract or Grant No. SPR-2144
12. Sponsoring Agency Name and Address Indiana Department of Transportation State Office Building 100 North Senate Avenue Indianapolis, IN 46204	13. Type of Report and Period Covered Final Report
	14. Sponsoring Agency Code

15. Supplementary Notes

Prepared in cooperation with the Indiana Department of Transportation and Federal Highway Administration.

16. Abstract

A highway system serves two needs: mobility and accessibility. Access control techniques are used to restrict access to the highway and improve vehicle flow. The objective of this research was to develop a comprehensive procedure to evaluate access control alternatives. The procedure includes the design and quantitative evaluation of alternatives to select the best one. Evaluation of each alternative includes prediction of turning volumes, delays, crash rates, and economic effectiveness. Several existing models predict traffic delays for signalized intersections and for minor streams at unsignalized intersections. Models are needed to predict delays of arterial streams caused by minor streams at unsignalized intersections. To address this missing component, models were developed to predict the delays caused to arterial streams by the following maneuvers: merging onto the arterial, diverging from the arterial, and left turn from the arterial. Models to predict crash rates for multi-lane arterial segments in Indiana based on geometric and access control characteristics were also developed. Models were developed to predict total, property-damage-only, and fatal/injury crashes. For the economic evaluation of each alternative, delays and stops are converted to operating costs for representative periods, and the crash rates are converted to crash costs. The agency costs can also be estimated. After the economic evaluation of each access control alternative, the best alternative can be selected.

17. Key Words	18. Distribution Statement
access control, urban arterial, user costs, unsignalized intersection, safety, transportation planning, highway management	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	176	

Form DOT F 1700.7 (8-69)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

IMPLEMENTATION REPORT

A procedure was developed to evaluate access control alternatives for high-speed urban arterials. The procedure is based on estimation of the access control impacts on safety, travel times, and fuel consumption. The steps of the procedure involve the design and analysis of access control alternatives. Evaluation of each alternative includes prediction of turning volumes, prediction of operating costs, prediction of crash costs, and prediction of agency costs. This procedure can be applied to existing arterials or to arterials in the design stage.

Both the INDOT Pre-Engineering and Environment Division and Planning and Programming Division will implement the procedure. The implementation will require learning several software tools such as TRANPLAN and TRANSYT and Excel spreadsheets that were developed as a part of the research. The purchase of other required software tools may be necessary as well. The principal investigator will be available for occasional assistance during this implementation process, which including the learning phase and incorporation into the agency's procedures, is expected to be labor and time consuming. It will involve resources external to the two above-mentioned INDOT divisions and may require collaboration with MPOs.

1. INTRODUCTION

1.1 Background

The functions of a highway system include providing motorists with mobility on highways and enabling access to surrounding lands. These two functions often conflict with each other. Restricting access can improve through vehicle flow by reducing disruptions caused by vehicles entering and exiting the highway. On the other hand, a road with frequent access points may be convenient for users trying to access adjacent properties but may cause frequent disruptions to through traffic. Because of this conflict between mobility and accessibility, the highway network has a hierarchical structure where different classes of roads have different functions. Freeways provide vehicles with maximum mobility, local roads provide maximum accessibility, and collectors and arterials provide intermediate levels of both mobility and accessibility.

Because arterials can be designed to provide some levels of both accessibility and mobility, a variety of access control levels are possible. An arterial can be designed with infrequent access points and other access control measures such as barrier medians to prevent left-turn and crossing maneuvers. An arterial can also be designed with a low level of access control with many commercial and residential driveways having direct access to the arterial.

The decision regarding what level of access control to implement can be difficult. The AASHTO geometric design policy (AASHTO, 1994) provides some general guidelines regarding access control, but more detailed and comprehensive guidelines are lacking. The decision to implement access control often involves engineering judgment as there are no uniform national guidelines regarding access control.

The decision to implement access control may be met with opposition by landowners and land users. Residents owning property adjacent to the arterial may be inconvenienced by loss of direct access to their property. Business owners may be concerned about potential reduced trade that could arise if access to their property is restricted. Customers doing business with commercial establishments may also be inconvenienced if access to these establishments is reduced. Disputes may develop because the needs of landowners and land users for accessibility to adjacent properties conflict directly with the mobility needs of highway users.

1.2 Problem Statement

A quantitative assessment of the benefits of access control is an important component of any access control decision-making process. An assessment of the benefits of access control measures is important for two reasons. First, it provides decision-makers with the opportunity to determine whether the potential user cost savings are justified by the capital investment in access control measures. Second, it provides highway agencies that represent highway users with supporting evidence for access control in potential disputes that may arise with landowners and land users. An assessment of access control benefits is especially important for high-speed urban arterials where minor stream traffic can create significant friction with the through traffic.

1.3 Objectives

The objective of this research is to develop a procedure to facilitate the access control decision-making process for high-speed urban arterials. The procedure evaluates economic effectiveness for various access control scenarios and can be applied to both existing and proposed arterials. The procedure is based on quantitative estimates of the costs and benefits of access control measures.

1.4 Scope of Work

In order to develop the procedure, impact models to predict the benefits of access control measures are needed. Some impact models already exist, while some other models needed to be developed. The existing and developed models can then be incorporated into an overall procedure.

1.5 Report Format

This report is organized as follows. Chapter 2 provides a summary of the literature review that was undertaken to develop an understanding of access control techniques and existing impact models. Chapter 3 describes the framework for the procedure to evaluate different access control alternatives. Chapter 4 discusses the traffic delay models that were developed in this study to help predict user costs and thus access control benefits. Chapter 5 concerns the development of the safety models for use in the procedure. Chapter 6 discusses the procedures for economic analysis to select the best access control alternative. Chapter 7 provides more details about the evaluation procedure and describes some implementation considerations. Chapter 8 contains a summary of the research findings and a description of the needs for future research.

2. LITERATURE REVIEW

The literature review consisted of several components. First, the existing access control techniques were reviewed. The initial list of access control techniques was then revised based on discussions with the Indiana Department of Transportation (INDOT). Existing delay models for unsignalized and signalized intersections were reviewed to determine what components could be used in the study and what components needed to be developed. Research studying the links between access control measures and traffic performance was also reviewed. Finally, safety models linking access control measures to crashes were studied.

2.1 Access Control Techniques

A literature review of existing access control techniques was undertaken. Azzeh et al. (1975) provide a list and brief description of 70 techniques. They define access control as “all techniques intended to minimize the traffic interference associated with commercial driveways.” They also classify these techniques based on similar applications and functional objectives. The application categories include highway design and operations techniques within the arterial traveled way, driveway location techniques aimed at limiting driveway number and location, and driveway design and operations techniques used outside of the arterial traveled way. Within these application categories, they group the techniques based on the following functional objectives: reducing conflict points, separating conflict areas, decreasing deceleration requirements, and separating turning vehicles from through vehicles.

The list of techniques given by Azzeh et al. (1975) was submitted to INDOT for revisions to reflect access control techniques that may be used in Indiana. These revisions led to the removal of some techniques and the addition of some techniques such as increasing turning radii and the use of median U-turns. These techniques were then further classified based on whether they influence the traffic pattern. A list of the techniques considered in this study and their effects on the traffic pattern is given in Appendix A. The results show that only certain techniques may affect the traffic pattern. For example, the prohibition of left turns influences the traffic pattern because vehicles wishing to turn left must find alternate ways to access the property. The addition of a right-turn deceleration lane does not influence the traffic pattern because it does not affect the manner in which vehicles access the driveway or local access point.

2.2 Existing Delay Models

A review of existing delay models in the literature was undertaken to determine what components could be used in the procedure and what components needed to be developed. Delay models for both unsignalized and signalized intersections were studied. In addition, several empirical and simulation studies that concerned the effects of access management on traffic performance were also reviewed.

2.2.1 Signalized Intersections

Well-developed tools already exist to assist engineers in analyzing signalized intersections. The *Highway Capacity Manual* (1994) provides a procedure to estimate delays and levels of service for signalized intersections. The *Highway Capacity Manual* is widely used in engineering practice. Tools also exist to assist engineers in the optimization of traffic signal settings. The PASSER II model (Wallace et al., 1991a) can be used to optimize the sequence of phases, cycle length, and phase lengths. The

TRANSYT-7F model (Wallace et al., 1991b) can optimize cycle length and phase lengths. The PASSER II and TRANSYT-7F models are primarily used for coordinated arterials but can also be applied to isolated intersections. The PASSER II model focuses on optimizing bandwidth. The TRANSYT-7F model is designed to optimize an objective function based on delays, stops, and overflow queues. The PASSER II and TRANSYT-7F models are integrated into a software package known as the Arterial Analysis Package (AAP) and are widely used in engineering practice.

Because of the availability of models and software packages to optimize and evaluate signalized intersections, additional components to estimate delays at signalized intersections did not need to be developed.

2.2.2 Unsignalized Intersections

The *Highway Capacity Manual* (1994) contains methods for estimating delays at unsignalized intersections. These methods are commonly used in current engineering practice. For all-way stop-controlled intersections, the *Highway Capacity Manual* can be used to estimate delays and levels of service for all approaches based on an empirical formula. For two-way stop-controlled intersections and T-intersections with a single minor street approach, the *Highway Capacity Manual* contains methods to estimate delays and levels of service for the minor streams caused by the priority streams as a function of minor stream volume, priority stream volume, and follow-up time. Delays can be calculated for vehicles crossing the major street, turning right onto the major street, turning left onto the major street, and turning left from the major street.

The *Highway Capacity Manual* does not contain a procedure to estimate delays for major (priority) streams at two-way stop-controlled intersections and T-intersections with a single minor street approach. Vehicles merging onto the arterial, diverging from the arterial, and turning left from the arterial can cause delays to through vehicles on the arterial. Literature pertaining to delay models for these various maneuvers was reviewed. The existing literature regarding the effects of minor streams on major streams focuses on the diverging and left-turn maneuvers.

2.2.2.1 Diverging Maneuver

Some past research efforts have included studies of the impacts of diverging maneuvers on major streams. Stover et al. (1970) used aerial time-lapse photographs and simulation to estimate the delays caused by diverging maneuvers based on driveway entrance speed and through lane volume. Their results may no longer be valid due to changes in drivers' behavior. Alexander (1970), studying mostly urban intersections, developed a multiple regression model to estimate the total delay per hour caused by vehicles diverging from two-lane streets. He found that the total delay caused by diverging vehicles was influenced by diverging volume, approach volume, and average speed of nondelayed through vehicles. His study was limited to sites with low to moderate volumes on two-lane highways.

One access control technique designed to reduce the impact of diverging vehicles on through vehicles involves the use of a right-turn deceleration lane. McCoy et al. (1994) used TRAF-NETSIM to develop regression models to estimate the delay savings provided by the right-turn deceleration lanes on two-lane and four-lane roadways. They found that the delay savings depended on diverging and through volumes. Some reservation should be given to the regression analysis of data obtained from simulation. The simulation models used to generate a database should first be scrutinized to check whether the investigated impacts are properly incorporated into these models.

For the purposes of evaluating access control techniques, analytical models to estimate delays caused by the diverging and merging maneuvers needed to be developed. A model was needed that could be used for both two-lane and multi-lane arterials and that could grasp subtle impacts such as changes in turning radii.

2.2.2.2 Left Turn from Major Street

Past research efforts devoted to developing guidelines for left-turn lanes have studied some aspects of the interaction between left-turning and through vehicles. Harmelink (1967) developed guidelines for left-turn lanes using queuing theory by

limiting the probability that a left-turning vehicle blocks the through lane for four-lane highways and by limiting the likelihood that a stopped left-turning vehicle blocks a through vehicle for two-lane highways. Ring and Carstens (1970) used multiple regression methods to estimate delays and stops for advancing vehicles as a function of advancing volume, opposing volume, and the proportion of left-turning vehicles. Their study was limited to rural highway intersections.

In some cases computer simulation has been used to study the influence of left-turning vehicles on through vehicles. Agent (1983) used the UTCS-1 Network Simulation Model to estimate the delay of the left-turn approach and used a critical delay level to develop guidelines for left-turn lanes. In a series of studies by McCoy et al. (1982), Ballard and McCoy (1983), McCoy et al. (1988), and Ballard and McCoy (1988), the General Purpose Simulation System Version H was used to simulate road sections with and without two-way left-turn lanes to estimate the operational benefits of these facilities to through traffic. Hawley and Stover (1996) used the TEXAS model for intersection traffic to determine the effect of left-turning vehicles on through vehicles at different speeds, advancing volumes, and opposing volumes. These simulation studies provide some insights into the benefits of left-turn lanes. However, the results of these studies may only be valid for the ranges of cases used in the simulation.

Recently Kyte et al. (1996) proposed a simple method to estimate the delays of through vehicles by assuming equal distribution of through traffic across the arterial lanes and probability of the left-turn queue occurrence. The shortcomings of this model are described by Bonneson and Fitts (1997), who point out that the model does not incorporate the effect of through vehicles changing lanes to reduce their delays. Bonneson and Fitts (1997) recently developed a model to estimate delays caused by left-turning vehicles using the equivalent factors method to deal with drivers' lane choice. They found that through-vehicle delay generally increased with increasing approach flow rate and left-turn percentage. In their model, the assumption of the drivers' behavior aimed at minimizing their travel times (or delays) when passing the intersection seems to be valid, but the solution can be questioned. Equalization of equivalent volumes across lanes is a simple and acceptable solution for traffic lanes that belong to the same lane

group when lane changing is not restricted. In the considered case, however, the lanes carrying uninterrupted streams generally experience delays lower than the lane carrying the left-turning vehicles. Thus, assuming that all drivers can change lanes in a timely manner, the leftmost lane should have no through vehicles at all. In real-world conditions, this situation is not observed as some through vehicles are trapped in the leftmost lane. Recognizing the above problems, Bonneson and Fitts introduce the concept of probability of a lane change, which unfortunately is postulated rather than derived from any stated assumptions.

To assist in the evaluation of access control alternatives, a simple procedure was developed to estimate delays caused by left-turning vehicles on arterial streets with several access points and typically no queues of vehicles turning left.

2.2.3 Overall Effects of Access Control on Traffic Performance

Several recent studies have investigated the effects of access control on vehicle speeds and travel times. Vargas and Reddy (1996) used TRAF-NETSIM to determine the effectiveness of proposed access control improvements for three arterials in Fort Lauderdale, Florida. They found that those improvements aimed at improving traffic flow resulted in better traffic performance, while safety oriented improvements did not significantly influence traffic flow. They concluded that locations should be analyzed carefully before access management improvement decisions are made. McShane et al. (1996) used TRAF-NETSIM to model effects of access management on traffic performance. They found that the addition of deceleration and left-turn lanes led to benefits for through traffic, while the addition of driveways led to reductions in arterial speeds. Hummer and Boone (1995) used TRAF-NETSIM to model the impacts of unconventional suburban arterial intersection designs, including the median U-turn. They found that the unconventional designs could improve traffic performance at some suburban arterial intersections.

In addition to the simulation studies, some empirical studies of the effects of access control on traffic performance have been undertaken. Garber and White (1996), using field data collected from 30 locations in Virginia, found that traffic density on urban minor arterials was influenced by average daily traffic per lane, average speed differential, right-turn lane availability, average driveway volumes, average driveway spacing, and average traffic signal spacing. Gattis (1996) measured travel times for 3 segments in Oklahoma and found that a segment with frequent access points had over 30 percent more delay than a segment with frontage roads. Poe et al. (1996) collected vehicle speed data from 34 sites on low-speed urban collector streets. They found that the frequency and location of access points influenced vehicle speeds. The results of these field studies may be influenced by local conditions.

2.3 Existing Safety Models

The existing literature regarding the influence of access control on safety was also reviewed. Some studies have looked at the overall influence of access management on safety. Other studies have focused on the effects of median treatments on safety. Some studies have been undertaken in Indiana.

2.3.1 Access Management and Safety

Past research efforts have included looking at the influence of access management on safety. Lall et al. (1996), studying a 29-mile corridor of the Oregon Coast Highway 9 (US 101), found a relationship between access density and the number of accidents for both rural and urban locations. Gattis (1996), in his study of 3 segments in a small Oklahoma city, found that the segment with the highest access control had accident rates approximately 40 percent lower than the other two segments. Garber and White (1996), in their study of 30 sections in Virginia, found that ADT per lane, average speed, number

of accesses, left-turn lane availability, average driveway spacing, and average difference in driveway spacing influenced the accident rate for urban principal arterials. Li et al. (1994), using a database of 163 sections of rural roads in British Columbia, studied the effects of geometric and access control characteristics on accidents. They classified access points into four major categories: unsignalized public road intersection, business access, residential access, and roadside pullout. They found a correlation between access density and accidents for all four categories with public road intersection density having the greatest influence on accidents. The results of these studies indicate that access control can help reduce the number of crashes. However, crash models based on sections in Indiana are desirable since local factors can influence safety.

2.3.2 Median Treatment

One common access control technique involves the use of medians to reduce the influence of left-turning vehicles on through vehicles. Some research has focused on the effects of various median treatments on safety. McCoy and Malone (1989) analyzed urban four-lane roadways in Nebraska to determine if the presence of left-turn lanes at signalized and uncontrolled approaches influenced accidents. They found that the presence of left-turn lanes led to a reduction in rear-end, sideswipe, and left-turn accident rates for both signalized and uncontrolled approaches. However, they also found that left-turn lanes led to an increase in right-angle accidents for uncontrolled approaches on urban undivided roadways. Squires and Parsonson (1989), studying four-lane and six-lane roadways in Georgia, developed regression equations to predict accidents for raised median and two-way left-turn lane sections. They found that raised medians generally had lower accident rates than two-way left-turn lanes except in cases where left turns were concentrated in a few areas on the road segment. McCoy et al. (1988) found that the accident rate on sections with a two-way left-turn lane was 34 percent lower than that on four-lane undivided sections. These models are focused on specific access control techniques related to median treatments.

2.3.3 Indiana Studies

Some studies have been undertaken to study the impacts of access control on safety in Indiana. McGuirk (1973) studied a total of 100 segments from 10 cities in Indiana. He found that the total number of driveways per mile and the number of commercial driveways per mile influenced the number of driveway accidents per mile. He reported that driveway accidents represented approximately 14 percent of all accidents on the sections. Uckotter (1974) studied 14 urban arterial segments from 5 cities in Indiana. He found that the driveway accident rate increased as the average daily driveway volume and average daily roadway volume increased. He also reported that 33 percent of the accidents on these sections were driveway accidents. The results of these studies indicate some of the factors that may influence driveway accidents in Indiana. However, models are needed to predict all types of crashes that may occur on arterial segments. In addition, driving conditions and drivers' behavior may have changed since these studies were undertaken.

In the most recent research by Eranky et al. (1997), crash reduction factors were developed for Indiana roads based on cross-sectional characteristics described in the Road Inventory Database. Separate negative binomial regression models were developed for rural two-lane, rural multi-lane, urban two-lane, and urban multi-lane highways. The model for urban multi-lane highways included the following variables: number of lanes, skid resistance factor, median width, median type-mountable, left-turn lane, presence of continuous left-turn lanes, outside shoulder width, inside shoulder width, and access control. A barrier type median was found to increase the number of accidents. The level of access control was described by a qualitative variable with three levels.

Based on the initial results of the study by Eranky et al. (1997), it was decided to further investigate the effects of access control on the number of accidents in Indiana. It was decided to incorporate variables such as the density of access points into a safety analysis. The results of this analysis could be used to predict accidents in Indiana based on the geometric and access control characteristics of road segments.

3. PROCEDURE TO EVALUATE ACCESS CONTROL ALTERNATIVES

This chapter describes the overall framework to evaluate access control alternatives developed as part of this study. The framework can be used for existing arterials and for arterials in the design stage. The procedure provides a framework for the user to develop and analyze different access control alternatives to select the best alternative.

3.1 Overall Framework

The evaluation of the effects of access control includes determining its impacts on safety, delays, and number of stops on the studied arterial and in the surrounding area (called the impact area) over a long period. The proposed method uses an economic approach in which benefits and costs are estimated. Figure 3.1 shows a flow chart of the overall procedure. The user first collects input data, designs the various alternatives, and determines the road network. Then the alternatives are evaluated. The analysis of each alternative begins with the prediction of turning volumes using transportation planning software such as TRANPLAN. Once turning volumes are predicted for each alternative, the impact area is determined. The impact area includes the region around the arterial in which the traffic pattern changes significantly between alternatives. Traffic delays, stops, and crash rates are then estimated for the impact area. The delays and stops are converted to operating costs, and the crash rates are converted to crash costs. The agency costs including construction and maintenance costs are also estimated for each alternative. Once each alternative has been evaluated, the user can select the best access control

alternative for implementation. The following sections describe some aspects of the procedure in greater detail.

3.2 Determine Road Network Representation

The road network is a collection of nodes and links between nodes that represent in a simplified way the actual system of streets. The road network representation should be sufficiently detailed to model in TRANPLAN those traffic streams that are required in the further steps. The required streams include turning movements at intersections and total number of vehicles exiting or entering the arterial at access points between the intersections as specified in Figure 3.2. Network nodes represent intersections. It is recommended that all signalized intersections in the impact area be represented as nodes because turning volumes are necessary input for the TRANSYT-7F model. It is assumed in the calculations that the access points do not experience significant queues. Thus, several access points can be represented by a single node. This simplification dramatically reduces the amount of effort required to prepare the road network representation. Of course, a fully-detailed network with all intersections and access points represented through individual nodes is also appropriate for the study.

3.3 Collect Input Data

The user cost is predicted for the lifetime of each alternative project. During this period (10-30 years) the following traffic variations are expected:

long-term but relatively slow changes caused by the region-wide trends and local gradual land development,

abrupt changes caused by the appearance of strong local traffic generators,

seasonal changes in an annual cycle,

day-to-day changes in a weekly cycle,

short-term changes in a daily cycle.

The abrupt changes in traffic pattern determine periods with slow traffic growth. Each such sub-period is represented by a mid-year. Alternatively, the first and final years of the sub-period can be used as representative years, and costs between these years can be estimated using linear interpolation. The representative year is represented by a typical weekday. Where it is justified and possible, also a weekend day can be considered. A typical day is divided into several intervals within which traffic volumes are assumed approximately steady. Each interval is represented by one hour. Thus, each sub-period is represented by several intervals of typical days of one or more representative years.

The input data must be sufficient to:

Define sub-periods with relatively slow general traffic growth,

Model network traffic in the representative hours for each sub-period,

Estimate delays, stops, and frequency of crashes for each network link.

More detailed specification of data needs can be found in the following sections of this report.

3.4 Predict Traffic Delays

In order to estimate operating costs, traffic delays need to be estimated. For each access control alternative, traffic delays are evaluated for the following traffic movements:

movements at intersections in the impact area using TRANSYT-7F,

arterial through movements at access points using the formulae developed in this study (explained below),

other movements at access points using a simplified *Highway Capacity Manual* method for unsignalized intersections.

The models developed in this study can be used to estimate the delays for the arterial streams caused at access points by merging, diverging, and left-turn maneuvers.

The input data requirements include traffic volumes, critical gaps, and average speeds. The results include the delay of arterial streams caused by minor streams in seconds per hour for each type of minor stream maneuver. The delays of the minor streams can also be calculated using a simplified *Highway Capacity Manual* method.

3.5 Predict Crash Rates

For each access control alternative, the crash rates are estimated using regression models and then converted to crash costs. The safety analysis of a given access control alternative focuses on the impact area. Crash rates for intersections and segments between intersections are estimated separately. A segment is a two-directional connector of two intersections. Typically, a segment includes two one-way links. In this study, regression models were developed to predict crash rates on arterial segments based on geometric and access control characteristics of the segments.

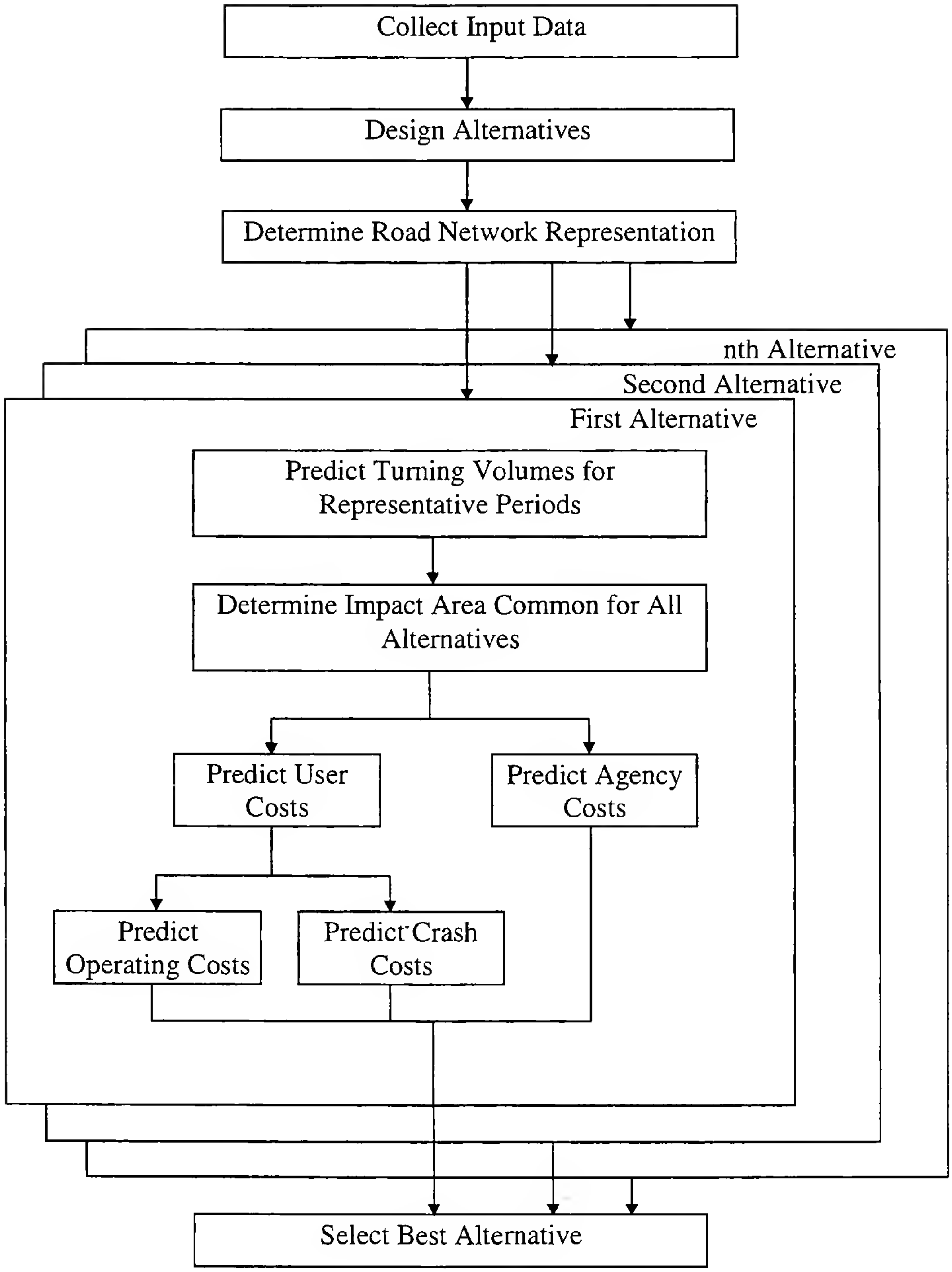

Figure 3.1 Framework for evaluation of access control alternatives

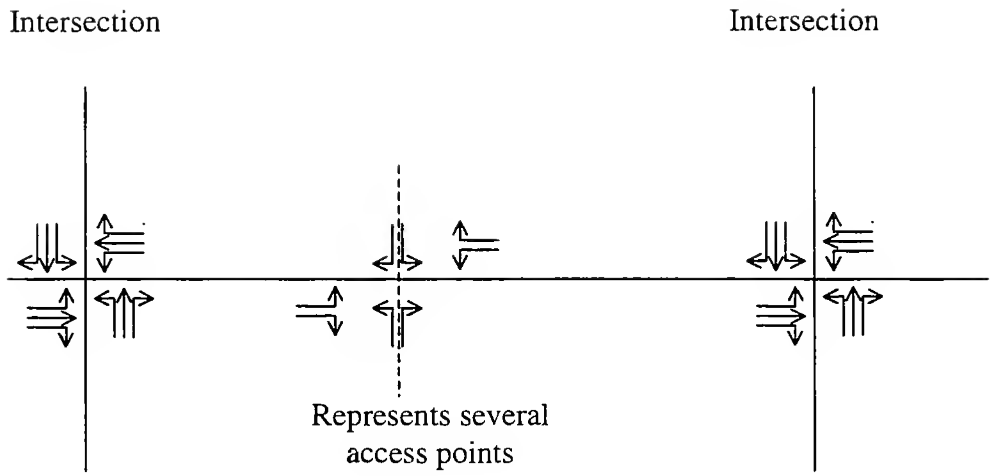

Figure 3.2 Traffic streams required for analysis

4. DELAY MODELS

Access points on the arterial are typically unsignalized. Although they cannot create bottlenecks along an arterial, their impact is spread along the arterial causing deterioration of the arterial performance. There are two major impacts of unsignalized access points: (a) their presence reduces free-flow speed of arterial vehicles, (b) the vehicles merging onto, diverging from, or turning left from the main street cause delays to the through vehicles. These two effects are of concern to traffic engineers trying to mitigate the adverse impacts of access points on the movement of the through vehicles. The *Highway Capacity Manual* (1994) provides a procedure for unsignalized intersections that estimates the capacities and delays for vehicles entering or crossing the major street and for vehicles turning left from the major street. The *Highway Capacity Manual* does not include a procedure to estimate the effects of the minor streams on the through vehicles. Thus, models to estimate the delays of through vehicles caused by minor streams needed to be developed as part of this study. Models were developed for three maneuvers: merging onto the arterial, diverging from the arterial, and turning left from the arterial. The models could then be incorporated into the procedure to evaluate access control alternatives.

The crossing maneuver is not considered for several reasons. First, crossing maneuvers from access points on an arterial typically are infrequent. Second, the crossing vehicle occupies a given traffic lane for a short period of time. Finally, the critical gap for the crossing maneuver is longer than for the merging maneuver. Thus, the crossing maneuver can be expected to have less impact on arterial vehicles than the other maneuvers.

4.1 Model Assumptions

The derivation of the models is based on the gap acceptance theory applied to vehicles on a lane-by-lane basis. The time headway in a given lane is assumed to follow a shifted negative exponential distribution:

$$f(g) = \frac{q \cdot s}{s - q} \cdot exp\left(\frac{g \cdot q \cdot s - q}{s - q} \right) \tag{4.1}$$

with cumulative density function

$$F(g) = 1 - exp\left(\frac{g \cdot q \cdot s - q}{s - q} \right) \tag{4.2}$$

where:

g = time headway between consecutive vehicles (sec),

q = flow rate in lane (veh/sec/lane),

s = maximum flow rate in lane (veh/sec/lane).

4.2 Impact Model for Merging Maneuver

A vehicle that merges onto the arterial from the crossing street may cause delay to the through vehicles on the arterial. The merging vehicle accepts a gap in the priority stream that is at least equal to the critical gap. After a lag time that includes decision and reaction time, the vehicle enters the arterial stream at an entry speed that is lower than the arterial speed. The entry speed can vary based on the approach geometry. An approach with a large turning radius or auxiliary lane may have a high entry speed. The distance traveled by the merging vehicle during the acceleration, s_m, for given entry and arterial speeds can be determined from a graph published by AASHTO (1994). The graph is partly based on a study by Olson et al. (1984). The acceleration rate and acceleration time for the merging vehicle can be calculated from the equations of motion, assuming a constant acceleration rate:

$$a_m = \frac{v_t^2 - v_m^2}{2 \cdot s_m} \tag{4.3}$$

$$t_m = \frac{v_t - v_m}{a_m} \tag{4.4}$$

where:

a_m = acceleration rate of merging vehicle (m/sec^2),

v_m = entry speed of merging vehicle (m/sec),

v_t = arterial speed (m/sec),

s_m = acceleration distance of merging vehicle (m),

t_m = acceleration time of merging vehicle (sec).

The value of the lag time was calibrated using field data from two unsignalized intersections. Merging vehicles were observed, and the lag time was recorded for each vehicle. The data were collected on June 18, 1997 in West Lafayette, Indiana. The first intersection was between a commercial driveway and a public road and had a relatively small radius of approximately 3 m. The second intersection was between two public roads and had a radius of approximately 7.5 m. Table 4.1 summarizes the results obtained. A test of the means indicated that the means were not significantly different from each other at a 0.05 level of significance. From these results, a value of 4 sec for lag time was assumed for the analysis. In the case of an access point with a right-turn acceleration lane, the lag time may be lower because drivers can begin to merge earlier. A value of 2 sec can be assumed when a full-length acceleration lane is present. An intermediate value between 2 sec and 4 sec can be assumed when a short acceleration lane is present.

Table 4.1 Results from lag time data collection

Location	Curb Radius (m)	Time of Observation	Mean Lag Time (sec)	Standard Deviation (sec)	Number of Observations
US 52 and McDonald's/KFC driveway	3	11:30 am-12:30 pm	4.00	0.95	29
Stadium Av. and Grant St.	7.5	3:50 pm-4:50 pm	3.72	0.61	85

4.2.1 Delay of First Through Vehicle

If the gap accepted by the merging vehicle is low, the first through vehicle behind the merging vehicle will decelerate to maintain a safe distance to the merging vehicle. The model assumes that when the merging vehicle reaches the arterial speed, the first through vehicle regains the arterial speed and trails the merging vehicle at the minimum safe distance. Figure 4.1 shows the time-space diagram for the merging maneuver used to derive the delay of the first through vehicle following the merging vehicle:

$$d_1 = t_m - \frac{s_m}{v_t} + g_r + \frac{1}{s} - g \text{, if } g < g_i, \tag{4.5}$$

$$= 0\text{, otherwise,}$$

where:

d_1 = delay of first through vehicle (sec),

t_m = acceleration time of merging vehicle (sec),

s_m = acceleration distance of merging vehicle (m),

v_t = arterial speed (m/sec),

g_r = lag time (sec),

g_i = impact gap above which there is no delay (sec).

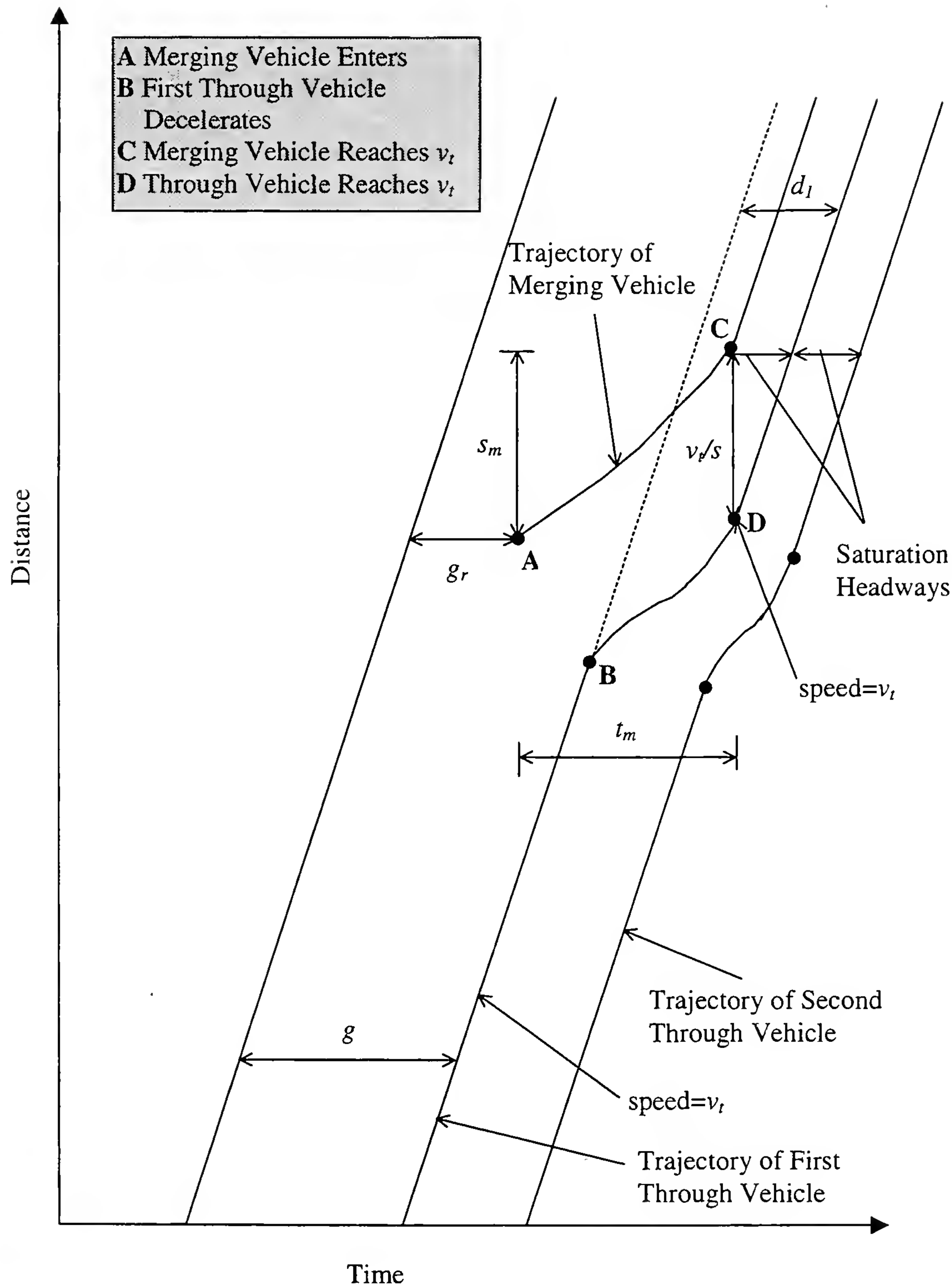

Figure 4.1 Time-space diagram for merging maneuver

If the gap g accepted by the merging vehicle is sufficiently large, the first through vehicle will not experience any delay. The impact gap g_i above which the through vehicle will not be influenced is given by

$$g_i = t_m - \frac{s_m}{v_t} + g_r + \frac{1}{s} \tag{4.6}$$

The average delay of the first vehicle before lane changing is considered can be found by integrating Equation 4.5 over the probability density function of g. The condition that a gap is larger than the critical gap must be incorporated since a vehicle will not merge if the gap is larger than the critical gap. Thus the average delay of the first vehicle before lane changing is incorporated is given by

$$E(d_{1,nlc}) = \int_{g_c}^{g_i} \frac{(g_i - g)\cdot f(g)dg}{1-F(g_c)} = g_i \cdot \frac{F(g_i)-F(g_c)}{1-F(g_c)} - \int_{g_c}^{g_i} \frac{g\cdot f(g)dg}{1-F(g_c)}$$

$$= g_i - g_c + \left(\frac{1}{q} - \frac{1}{s}\right)\cdot\left(exp\left(-\frac{q\cdot s\cdot (g_i - g_c)}{s-q}\right) - 1\right) \tag{4.7}$$

where:

$E(d_{1,nlc})$ = average delay of first through vehicle before lane changing is incorporated (sec),

g_c = critical gap for merging (sec).

For multi-lane arterials, the through vehicle may try to avoid delay by changing lanes. Assuming that a vehicle changing lanes experiences no delay, the effect of lane changing can be incorporated by multiplying the average delay by 1-P_s, where P_s is the likelihood that a through vehicle changes lanes. The likelihood of changing lanes depends on the availability of sufficient space gaps in the adjacent lane and on drivers' preferences. The likelihood of a lane change is given by

$$P_s = P_a \cdot P_g$$

$$= P_a \cdot exp\left(-\frac{h_1 \cdot s \cdot q_a - v_t \cdot q_a}{v_t \cdot (s - q_a)} \right) \tag{4.8}$$

where:

P_s = likelihood that a through vehicle changes lanes,

P_a = likelihood that a driver wants to change lanes,

P_g = likelihood that there is sufficient space to change lanes,

q_a = flow rate in adjacent lane (veh/sec/lane),

h_l = critical space gap for lane changing (m), typically 4 sec × v_t.

After incorporating lane changing, the overall average delay of the first vehicle caused by one merging maneuver is given by

$$E(d_1) = (1 - P_s) \cdot E(d_{1,nlc}) \tag{4.9}$$

where:

$E(d_l)$ = average delay of first through vehicle (sec).

4.2.2 Delay of Other Through Vehicles

The merging vehicle may influence more than one vehicle on the arterial stream. Once the delay of the first through vehicle is known, the delay for other vehicles can be determined. Based on the assumption that time headways between consecutive vehicles follow a shifted negative exponential distribution, it can be derived that the first through vehicle and a trailing kth vehicle ($k > 1$) are separated by a total time headway that follows a shifted gamma distribution given by

$$f(h) = \frac{\left(h - \frac{n}{s} \right)^{k-2} \cdot \left(\frac{q \cdot s}{s - q} \right)^{k-1} \cdot exp\left(-\frac{h \cdot s \cdot q - (k-1) \cdot q}{s - q} \right)}{(k-2)!} \tag{4.10}$$

where h is the total headway between kth vehicle and first vehicle (sec).

The trailing vehicle will use any available headway in excess of the minimum one to reduce its delay. The minimum headway between the first and *k*th trailing vehicle after full compression of the traffic beyond the merging vehicle is *(k-1)/s*. If some vehicles between the merging vehicle and trailing vehicle change lanes, the number of required consecutive saturation headways is reduced, thus the minimum headway is *(k-1-l)/s*, where *l* is the number of vehicles between the merging and *k*th through vehicles that manage changing lanes. Thus the delay of the *k*th vehicle is given by

$$d_k = d_1 - h + \frac{(k-1-l)}{s}, \quad \text{if } h < h_i, \tag{4.11}$$
$$= 0, \quad \text{otherwise,}$$

$$h_i = d_1 + \frac{(k-1-l)}{s} \tag{4.12}$$

where:

d_1 = delay of first through vehicle (sec),

l = number of vehicles between the merging and *k*th through vehicles that manage changing lanes,

h_i = impact total headway (sec).

The values of d_1, l, and h vary randomly according to their distributions. Finding the average delay of vehicle k thus involves integrating Equation 4.11 with respect to the probability density functions of d_1, l, and h. An approximation was used in which d_1 and l were replaced with their average values. Other through vehicles may experience delay even if the first vehicle changes lanes, so the average value of the delay of the first vehicle in the approximation does not include lane changing. The average value of l is given by

$$E(l) = (k-1) \cdot P_s \tag{4.13}$$

where:

P_s = likelihood that a through vehicle changes lanes.

The average delay of vehicle k can be estimated by integrating Equation 4.11 with respect to the probability density function of h from $(k-1)/s$ to h_i. Thus, the average delay of vehicle k is given by

$$E(d_k) = (1-P_s)\cdot\left(\begin{aligned} &exp\left(-\frac{C\cdot s\cdot q}{s-q}\right)\cdot\left(\frac{(k-1)\cdot(s-q)}{s\cdot q}+A-C\cdot B-C\right) \\ &+C-\frac{(k-1)\cdot(s-q)}{s\cdot q}\end{aligned}\right) \tag{4.14}$$

$$A = \sum_{i=1}^{k-1}\frac{(s\cdot q)^{k-1-i}\cdot(k-1)\cdot C^{k-i}}{(s-q)^{k-1-i}\cdot(k-i)!} \tag{4.15}$$

$$B = \sum_{i=1}^{k-2}\frac{(s\cdot q)^{k-1-i}\cdot C^{k-1-i}}{(s-q)^{k-1-i}\cdot(k-1-i)!}\text{, if } k>2, \tag{4.16}$$

$$= 0\text{, otherwise,}$$

$$C = E(d_{1,nlc}) - \frac{E(l)}{s} \tag{4.17}$$

4.2.3 Average Delay Caused by Merging Maneuver

The average delay caused by one merging maneuver can be found by adding the average delay of the first through vehicle $E(d_1)$ to the average delay of the other through vehicles. The total delay caused by merging maneuvers at a given location can be found by multiplying the average delay caused by one merging maneuver by the number of merging maneuvers.

Table 4.2 Parameters used for sensitivity analysis for merging and diverging models

Parameter	Value
s	0.5 veh/sec
g_r	4 sec
g_c	5.5 sec
h_l	(4 sec) × v_t
t_r	1.5 sec
b_m	-4.9 m/sec^2
a_t	1.5 m/sec^2

4.2.4 Sensitivity Analysis for Merging Maneuver

Figures 4.2 to 4.5 show sample results from a sensitivity analysis for the average delay caused by one merging maneuver. The parameters used are listed in Table 4.2. An equal traffic distribution between lanes was assumed. Each of the four graphs corresponds to a different combination of values for the likelihood that a driver wants to change lanes (P_a) and merging speed (v_m). Within each graph, the arterial volume and arterial speed (v_t) vary. The results show that for a given arterial volume, the average delay increases as the arterial speed increases. For a given arterial speed, the average delay increases as the arterial volume increases. The effect of increasing volume is nonlinear. For high volumes, the average delay of the first through vehicle is high, and many vehicles behind the first through vehicle also experience delays.

A comparison between the graphs shows the effects of lane changing and entry speed. As the entry speed increases, the average delay caused by a merging maneuver decreases. Thus, an increased turning radius or right-turn deceleration lane can help reduce delay by increasing the entry speed of the merging vehicle. The graphs also indicate the effects of lane changing. The graphs corresponding to a P_a value of zero indicate cases with no drivers who attempt to change lanes, while the graphs corresponding to a P_a value of 0.5 indicate cases where 50 percent of drivers will attempt to change lanes. The results show that lane changing can have a significant effect on the delay caused by merging. Thus for multi-lane arterials, the delay caused by merging will

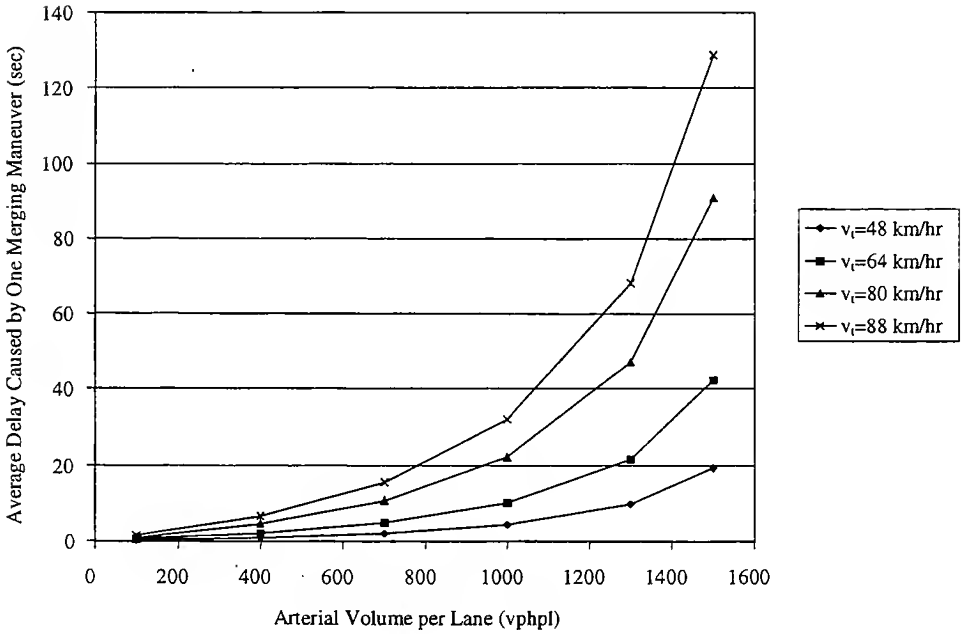

Figure 4.2 Delay caused by merging for P_a=0 and v_m=10 km/hr

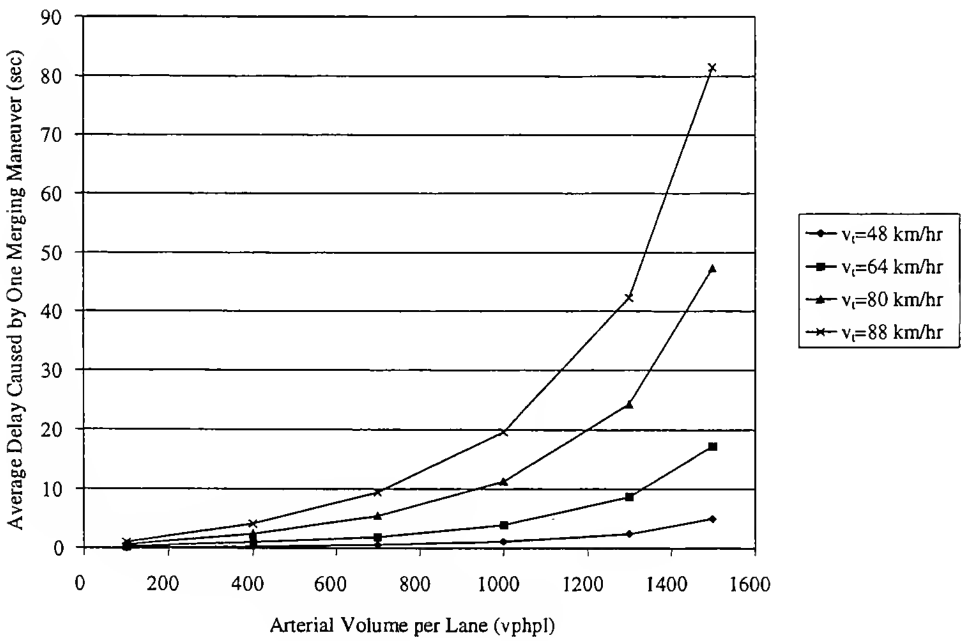

Figure 4.3 Delay caused by merging for P_a=0 and v_m=20 km/hr

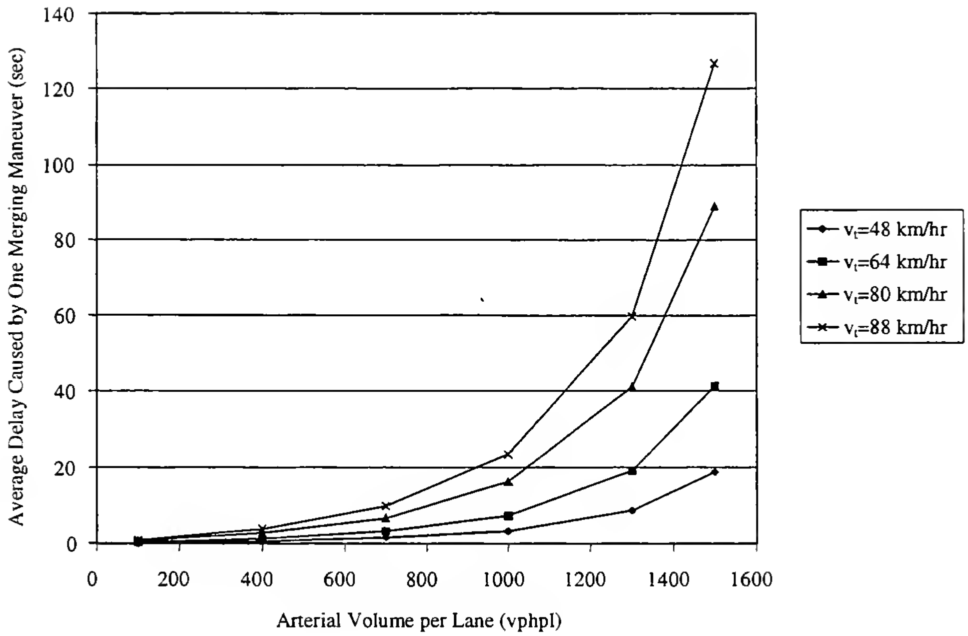

Figure 4.4 Delay caused by merging for P_a=0.5 and v_m=10 km/hr

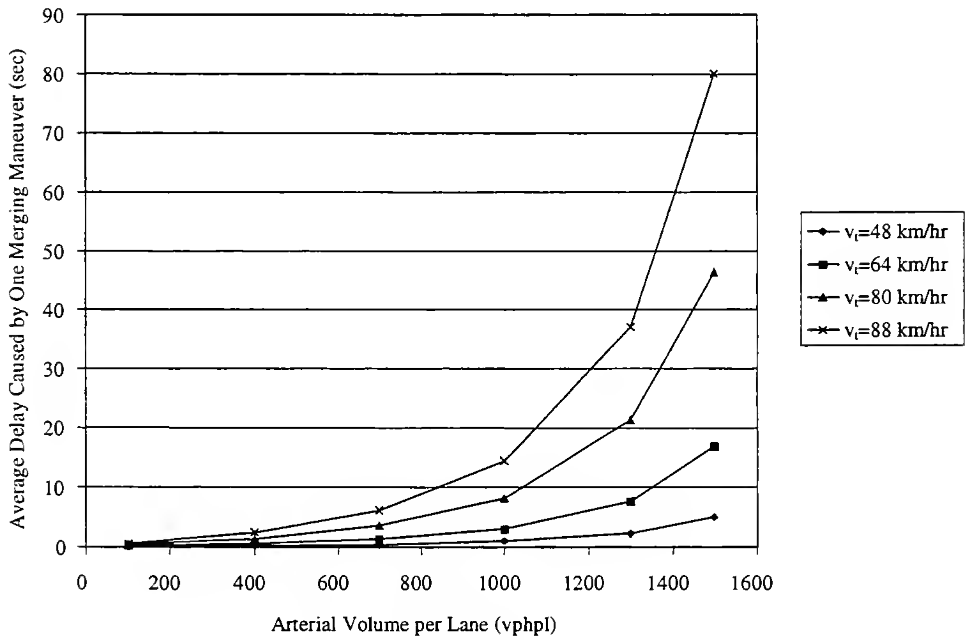

Figure 4.5 Delay caused by merging for P_a=0.5 and v_m=20 km/hr

be lower than for two-lane arterials. However, the effect of lane changing becomes diminished at high arterial volumes. As the arterial volume becomes high, lane changing becomes infrequent regardless of the portion of drivers who try to do so because most of the space gaps inspected in the adjacent lane will be too short.

4.3 Impact Model for Diverging Maneuver

The general procedure to estimate the delay caused by the diverging maneuver is similar to that for the merging maneuver. However, the diverging maneuver is different from the merging maneuver in some aspects. First, there is no gap acceptance process. A vehicle on the arterial will diverge regardless of the headway between it and the following vehicle. In addition, the speed of the through vehicle at the moment when the diverging vehicle exits needs to be estimated. If this speed is less than the arterial speed, the through vehicle will experience delay.

Similar to the merging maneuver, the model requires arterial and diverging speeds as input. The diverging speed again will vary based on intersection geometry. The deceleration distance for the diverging vehicle, s_d, can be estimated from graphs provided by AASHTO (1994) for given arterial and diverging speeds. The deceleration rate and deceleration time for the diverging vehicle can be calculated from the equations of motion, assuming a constant deceleration rate:

$$b_d = \frac{v_d^2 - v_t^2}{2 \cdot s_d} \tag{4.18}$$

$$t_d = \frac{v_d - v_t}{b_d} \tag{4.19}$$

where:

b_d = deceleration rate of diverging vehicle (m/sec^2),

v_d = exit speed of diverging vehicle (m/sec),

v_t = arterial speed (m/sec),

s_d = deceleration distance of diverging vehicle (m),

t_d = deceleration time of diverging vehicle (sec).

The delay of the first through vehicle depends on the rate at which the through vehicle decelerates.

4.3.1 Deceleration at Maximum Rate

In some rare cases where the initial headway between the diverging vehicle and the first through vehicle is low, the through vehicle's deceleration may be limited by the maximum deceleration rate, b_m. Under these circumstances, the first through vehicle will not be able to maintain the safety headway *1/s*.

When the first through vehicle is limited by the maximum allowable deceleration rate, the through vehicle decelerates at the constant rate b_m until the diverging vehicle exits. After the diverging vehicle exits, the through vehicle will experience an acceleration delay until it reaches the arterial speed. During the deceleration phase, the through vehicle experiences a loss in distance during the deceleration time given by

$$\Delta s_{b,a} = x_1 - x_2 \tag{4.20}$$

$$x_1 = v_t \cdot (t_d - t_r) \tag{4.21}$$

$$x_2 = v_t \cdot (t_d - t_r) + \frac{1}{2} \cdot b_m \cdot (t_d - t_r)^2 \tag{4.22}$$

where:

$\Delta s_{b,a}$ = loss in distance during deceleration phase for maximum deceleration (m),

x_1 = distance traveled in $(t_d - t_r)$ under no delay for maximum deceleration (m),

x_2 = distance traveled in $(t_d - t_r)$ under delay for maximum deceleration (m),

v_t = arterial speed (m/sec),

t_d = deceleration time of diverging vehicle (sec),

t_r = reaction time of through vehicle (sec),

b_m = maximum allowable deceleration rate (m/sec^2).

After the diverging vehicle exits, the through vehicle has a speed given by

$$v_{b,a} = v_t + b_m \cdot (t_d - t_r) \tag{4.23}$$

where:

$v_{b,a}$ = speed of first through vehicle when diverging vehicle exits under maximum deceleration (m/sec).

The through vehicle then experiences an acceleration delay until it reaches the arterial speed. The loss in distance incurred in reaching the arterial speed is given by

$$\Delta s_a = x_3 - x_4 \tag{4.24}$$

$$x_3 = v_t \cdot t_a = v_t \cdot \left(\frac{v_t - v_{b,a}}{a_t} \right) \tag{4.25}$$

$$x_4 = \frac{v_t^2 - v_{b,a}^2}{2 \cdot a_t} \tag{4.26}$$

where:

Δs_a = loss in distance during acceleration phase (m),

a_t = acceleration rate of through vehicle after diverging vehicle exits (m/sec^2),

t_a = time to accelerate from $v_{b,a}$ to v_t (sec),

x_3 = distance traveled at constant arterial speed during time t_a (m),

x_4 = distance traveled while accelerating from $v_{b,a}$ to v_t (m).

The delay of the first vehicle is then found as the total loss in distance during deceleration and acceleration divided by the arterial speed:

$$d_{1,a} = \frac{\Delta s_{b,a} + \Delta s_a}{v_t} = -\frac{b_m \cdot (t_d - t_r)^2}{2 \cdot v_t} + \frac{(v_t - v_{b,a})^2}{2 \cdot v_t \cdot a_t} \tag{4.27}$$

where $d_{1,a}$ is the delay of first vehicle under maximum deceleration (sec).

Assuming that the values in Equation 4.27 are fixed for given arterial and diverging speeds, Equation 4.27 also represents the average delay of the first through vehicle under maximum deceleration $E(d_{1,a})$.

This case of maximum deceleration occurs when the initial gap between the diverging and first through vehicle is less than g_s given by

$$g_s = MAX\left(\frac{1}{s}, \frac{b_m \cdot \left[\frac{2 \cdot (t_d - t_r)}{s} + (t_d - t_r)^2 \right] - 2 \cdot s_d}{2 \cdot v_t} + \frac{1}{s} + t_d \right) \tag{4.28}$$

where:

g_s = gap below which maximum deceleration occurs (sec).

The case of maximum deceleration occurs with likelihood

$$P_{i,a} = P(g < g_s) = 1 - exp\left(-\frac{g_s \cdot q \cdot s - q}{s - q} \right) \tag{4.29}$$

where:

$P_{i,a}$ = likelihood of braking at maximum deceleration rate.

Typically the case of braking at maximum deceleration rate is not likely to occur.

4.3.2 Deceleration at Rate Less Than Maximum

When the first through vehicle does not need to use the maximum deceleration rate, the deceleration rate depends on the initial headway between the diverging vehicle and the first through vehicle. As for the case of maximum deceleration rate, the speed of the through vehicle when the diverging vehicle leaves the arterial needs to be estimated.

If this speed is less than the arterial speed, the through vehicle will experience an acceleration delay until it reaches the arterial speed. The model assumes that for short headways the first through vehicle will apply a constant rate of deceleration after an initial reaction time. At the time of diverging, the first through vehicle must be at a safety distance to the diverging vehicle. Figure 4.6 shows the time-space diagram for the diverging maneuver. Under these assumptions, the loss in distance during the deceleration phase is given by

$$\Delta s_{b,b} = x_5 - x_6 \tag{4.30}$$

$$x_5 = v_t \cdot (t_d - t_r) \tag{4.31}$$

$$x_6 = g \cdot v_t + s_d - \frac{v_{b,b}}{s} - t_r \cdot v_t \tag{4.32}$$

where:

$\Delta s_{b,b}$ = loss in distance during deceleration phase for unrestricted deceleration (m),

x_5 = distance traveled in (t_d-t_r) under no delay for unrestricted deceleration (m),

x_6 = distance traveled in (t_d-t_r) under delay for unrestricted deceleration (m),

v_t = arterial speed (m/sec),

t_d = deceleration time of diverging vehicle (sec),

t_r = reaction time of through vehicle (sec),

g = initial headway between diverging vehicle and first through vehicle (sec),

s_d = deceleration distance of diverging vehicle (m),

$v_{b,b}$ = speed of through vehicle when diverging vehicle exits arterial (m/sec).

The through vehicle then accelerates until it reaches the arterial speed. The loss in distance during the acceleration phase can be calculated in the same manner as for the case of maximum deceleration after substituting $v_{b,b}$ for $v_{b,a}$. The delay of the first through vehicle is then the total loss in distance divided by the arterial speed:

$$d_{1,b} = \frac{\Delta s_{b,b} + \Delta s_a}{v_t}$$

$$= t_d - g - \frac{s_d}{v_t} + \frac{v_{b,b}}{v_t \cdot s} + \frac{(v_t - v_{b,b})^2}{2 \cdot v_t \cdot a_t}, \text{if } g < g_i, \tag{4.33}$$

=0, otherwise,

$$g_i = t_d + \frac{1}{s} - \frac{s_d}{v_t} \tag{4.34}$$

where g_i is the impact gap above which there is no delay (sec).

To calculate the delay of the first vehicle in this case, the speed of the first through vehicle when the diverging vehicle exits needs to be calculated. This value can be calculated using the equations of motion:

$$v_{b,b} = v_t + b_t \cdot (t_d - t_r) \tag{4.35}$$

$$v_{b,b}^2 = v_t^2 + 2 \cdot b_t \cdot x_6 \tag{4.36}$$

Thus, the deceleration rate of the first through vehicle and speed of the first through vehicle when the diverging vehicle exits are given by

$$b_t = \frac{2 \cdot v_t \cdot \left(g - \frac{1}{s} - t_d \right) + 2 \cdot s_d}{\frac{2}{s} \cdot (t_d - t_r) + (t_d - t_r)^2}, \text{if } g < g_i, \tag{4.37}$$

= 0, otherwise,

$$v_{b,b} = v_t + \frac{2 \cdot v_t \cdot \left(g - \frac{1}{s} - t_d \right) + 2 \cdot s_d}{\frac{2}{s} + t_d - t_r}, \text{if } g < g_i, \tag{4.38}$$

= v_t, otherwise,

where:

b_t = deceleration rate of through vehicle (m/sec^2),

g = initial headway between diverging vehicle and first through vehicle (sec),

t_r = reaction time of through vehicle (sec),

g_i = impact gap above which there is no delay (sec),

$v_{b,b}$ = speed of first through vehicle when diverging vehicle exits arterial (m/sec).

The likelihood of a gap between g_s and g_i and conditional average gap value are given by

$$P_{i,b} = P(g_s < g < g_i) = exp\left(-\frac{g_s \cdot q \cdot s - q}{s - q}\right) - exp\left(-\frac{g_i \cdot q \cdot s - q}{s - q}\right) \tag{4.39}$$

$$E(g|g_s < g < g_i) = \frac{\int_{g_s}^{g_i} g \cdot f(g) dg}{F(g_i) - F(g_s)}$$

$$= \frac{exp\left(-\frac{g_s \cdot q \cdot s - q}{s - q}\right) \cdot \left(g_s + \frac{1}{q} - \frac{1}{s}\right) - exp\left(-\frac{g_i \cdot q \cdot s - q}{s - q}\right) \cdot \left(g_i + \frac{1}{q} - \frac{1}{s}\right)}{exp\left(-\frac{g_s \cdot q \cdot s - q}{s - q}\right) - exp\left(-\frac{g_i \cdot q \cdot s - q}{s - q}\right)} \tag{4.40}$$

where:

$P_{i,b}$ = likelihood of gap between g_s and g_i,

$E(g\ /g_s<g<g_i)$ = conditional average gap value between g_s and g_i (sec).

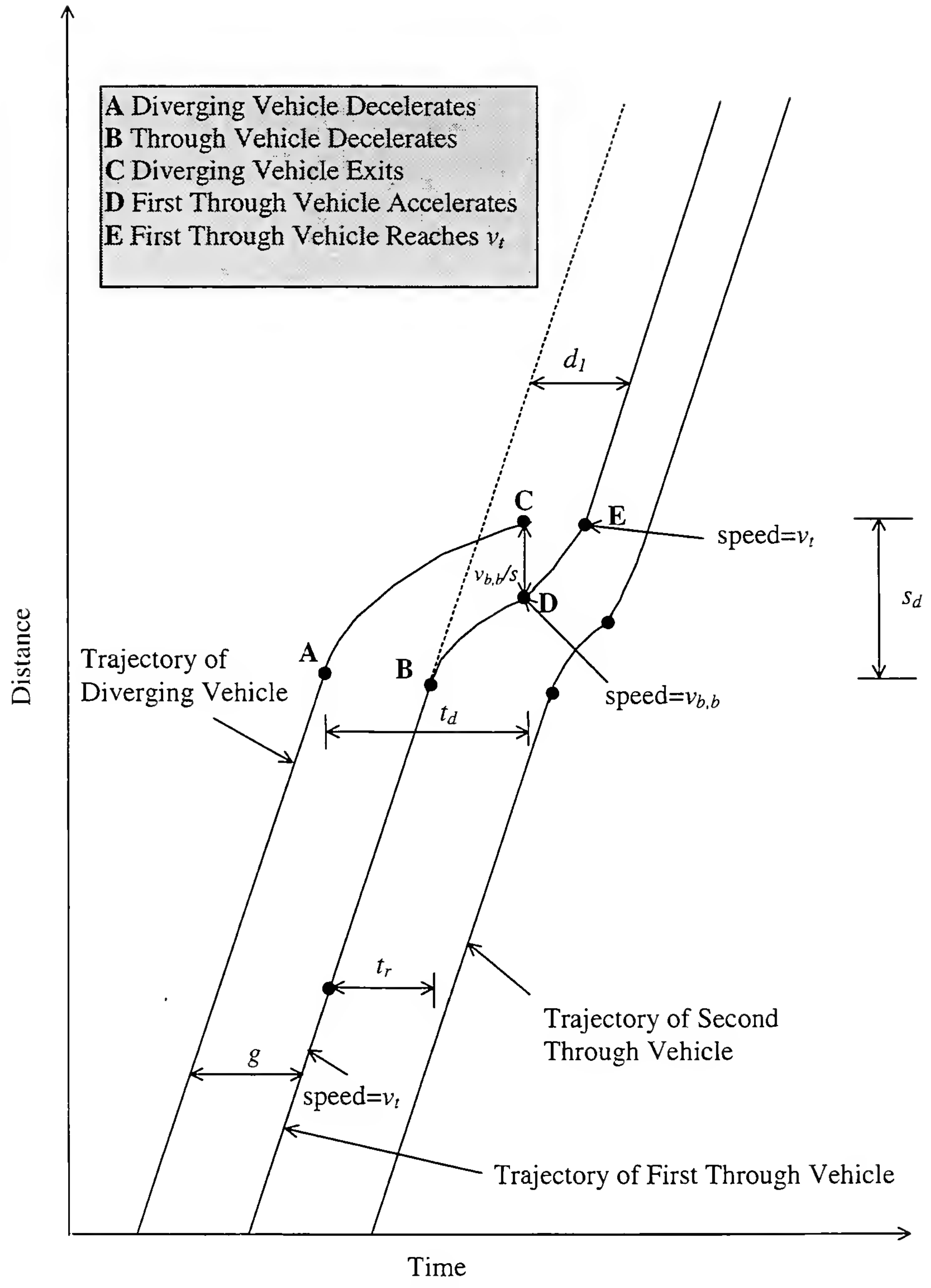

Figure 4.6 Time-space diagram for diverging maneuver (deceleration rate less than maximum)

The conditional average delay of the first vehicle at a deceleration rate less than maximum $E(d_{1,b}/d_{1,b}>0)$ can be found by substituting the average gap value from Equation 4.40 into Equations 4.33 and 4.38.

4.3.3 Average Delay Caused by Diverging Maneuver

The average delay of the first through vehicle without lane changing can be determined by combining the cases of deceleration at the maximum rate and deceleration at a rate less than maximum. The average delay of the first vehicle before lane changing is incorporated is given by

$$E(d_{1,nlc}) = P_{i,a} \cdot E(d_{1,a}) + P_{i,b} \cdot E(d_{1,b} | d_{1,b} > 0) \tag{4.41}$$

As for the merging case, lane changing can also be incorporated. Assuming that a vehicle changing lanes experiences no delay, the overall average delay of the first through vehicle caused by one diverging maneuver is given by

$$E(d_1) = (1 - P_s) \cdot E(d_{1,nlc}) \tag{4.42}$$

Once the average delay of the first through vehicle has been estimated, the average delay for the other vehicles can be estimated using the same procedure as for the merging case. The average delay caused by one diverging maneuver can then be found by adding the average delay of the first through vehicle $E(d_1)$ to the average delay of the other vehicles. The total delay caused by diverging maneuvers can be found by multiplying the average delay caused by one diverging maneuver by the number of diverging maneuvers.

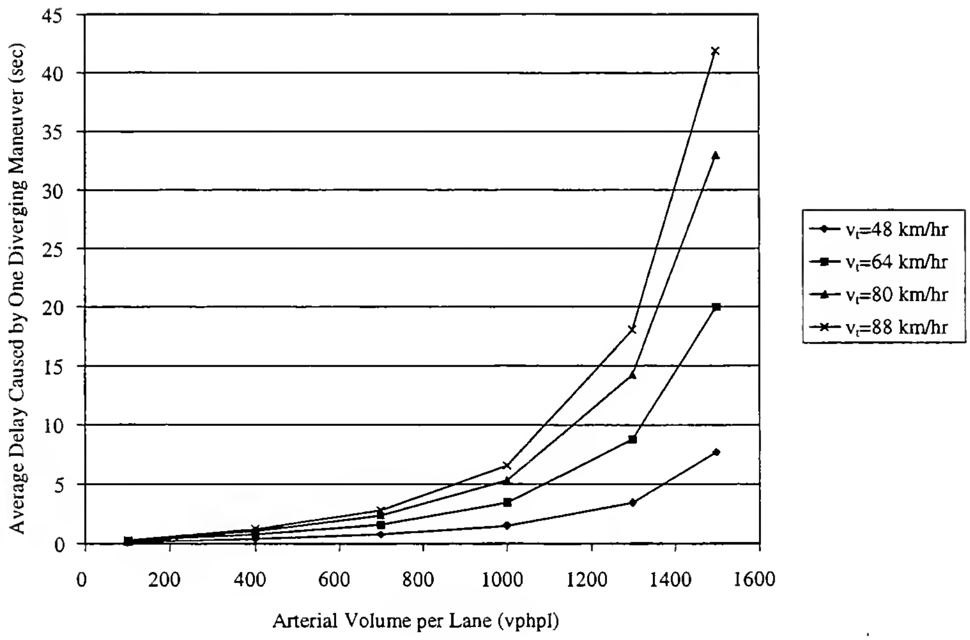

Figure 4.7 Delay caused by diverging for P_a=0 and v_d=10 km/hr

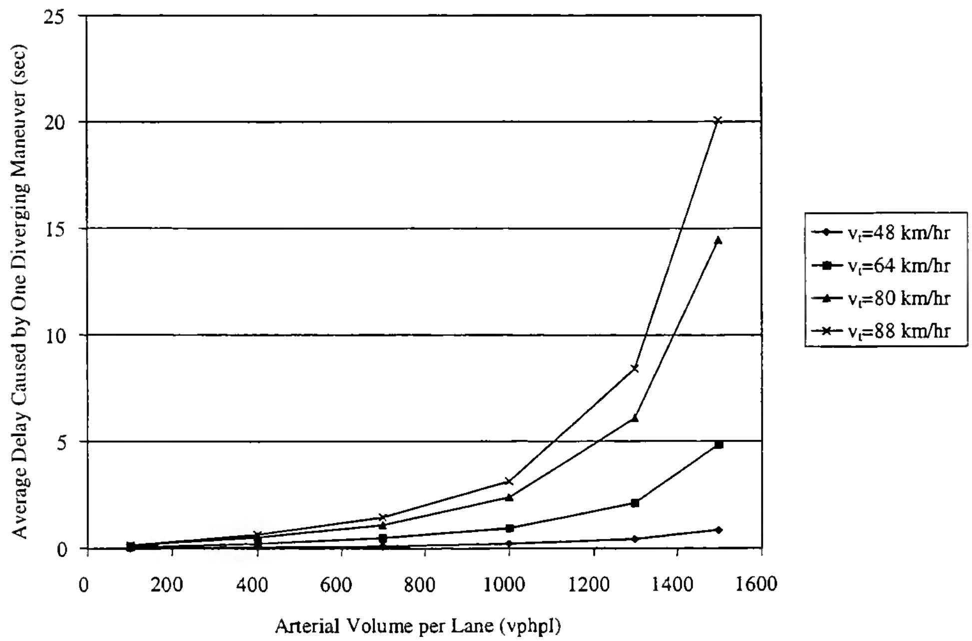

Figure 4.8 Delay caused by diverging for P_a=0 and v_d=20 km/hr

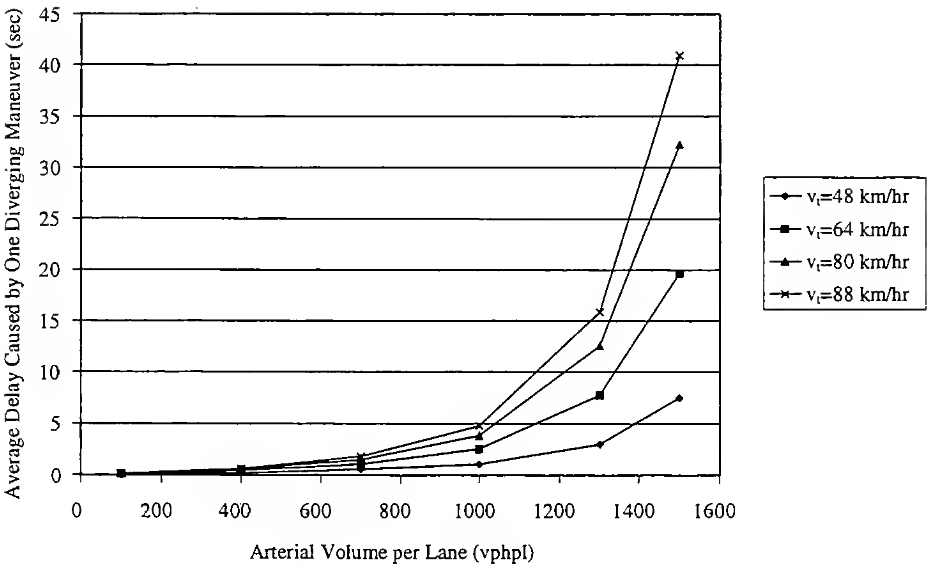

Figure 4.9 Delay caused by diverging for P_a=0.5 and v_d=10 km/hr

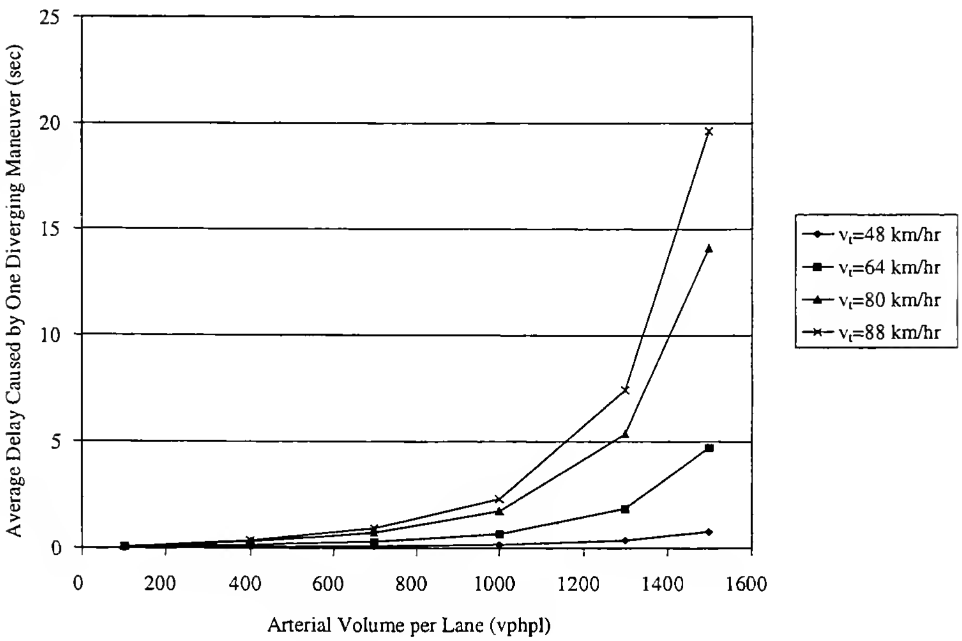

Figure 4.10 Delay caused by diverging for P_a=0.5 and v_d=20 km/hr

4.3.4 Sensitivity Analysis for Diverging Maneuver

Figures 4.7 to 4.10 show sample results from a sensitivity analysis of the average delay caused by one diverging maneuver. As for the merging maneuver, an equal traffic distribution between lanes was assumed. The results indicate trends similar to those for the merging maneuver. Average delay increases as the arterial volume and arterial speed increase. The exit speed of the diverging vehicle and the drivers' preferences for changing lanes appear to influence the delay caused by one diverging maneuver. The effect of lane changing becomes less significant at high arterial volumes.

4.4 Impact Model for Left Turn from Major Street

Vehicles turning left from the arterial can cause delay to through vehicles, especially when the left-turning and through vehicles share a lane on the arterial. A left-turning vehicle in a shared lane blocks through vehicles as it waits for a sufficient gap in the opposing stream. In order to minimize the effects of left-turning vehicles on through vehicles, exclusive left-turn lanes may be provided. To estimate the operational benefits of an exclusive left-turn lane, the delays caused by the left-turning vehicles need to be estimated.

4.4.1 Assumptions for Left-Turn Model

Vehicles may turn left from an arterial segment at many different unsignalized access points along the segment. Because each individual access point is not represented in the network, the turning volumes available will typically include the total volume turning left from the arterial segment in each direction. The following assumptions are made about left turns at individual access points:

At each access point, there is no queuing of left-turning vehicles.

The effect of the blockage caused by a left-turning vehicle at a given access point is independent of the blockage effect at another access point.

Under these assumptions, the time during which a left-turning vehicle blocks the shared lane while waiting for the gap in the opposing stream is given by

$$t_l = \frac{3600}{c_l} \tag{4.43}$$

where:

t_l = average blockage time of left-turning vehicle (sec),

c_l = capacity of left-turning movement calculated from *Highway Capacity Manual* (veh/hr).

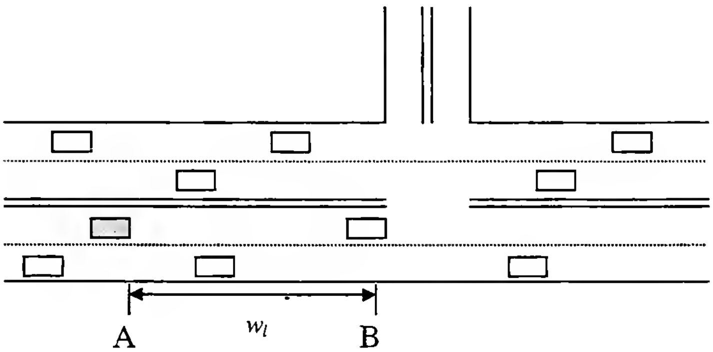

Figure 4.11 Through vehicle approaching left-turning vehicle

4.4.2 Effect of Lane Changing

For multi-lane arterials, through vehicles in the shared lane may attempt to change lanes when they see a blocking left-turn vehicle. Figure 4.11 shows the situation as a through vehicle approaches a left-turning vehicle. At distance w_l (point A), the left-turning vehicle has become obvious to the through driver. The driver changes lanes if there is a sufficient opening in the stream in the adjacent lane. If not, the driver starts

seeking a sufficient space in the adjacent stream by maintaining speed lower by value Δv from the speed in the adjacent lane v_t. The driver is not able to change lanes if all the inspected spaces are too short. The probability of such an event is

$$P_j = (1 - P_{s2})^r \tag{4.44}$$

where:

P_j = likelihood that a through vehicle is unable to change lanes in advance of a blocking left-turning vehicle,

P_{s2} = probability of accepting a single gap,

r = number of gaps rejected.

The probability P_{s2} can be estimated using a similar approach as was used in Equation 4.8. A vehicle will be able to change lanes if the driver prefers lane changing and if a given inspected space gap in the adjacent lane is sufficiently long. The proportion of drivers who will try to change lanes to avoid delay from a left-turning vehicle may be different than the proportion of drivers who will try to change lanes to avoid delay from merging and diverging. The critical space gap for lane changing may also be different than for the diverging and merging cases. Thus, the probability of accepting a given gap is

$$P_{s2} = P_{a2} \cdot exp\left(-\frac{h_2 \cdot q_a \cdot s - v_t \cdot q_a}{v_t \cdot (s - q_a)} \right) \tag{4.45}$$

where:

h_2 = critical space headway for in-advance lane change (m),

q_a = flow rate in the adjacent lane (veh/sec/lane),

P_{a2} = likelihood that a driver wants to change lanes in advance of left-turning vehicle.

The driver of the through vehicle can inspect gaps in the adjacent lane as he approaches the left-turning vehicle. The total time the driver spends inspecting gaps in the adjacent lane is the time it takes the through vehicle to travel the distance w_l:

$$t = \frac{w_l}{v_t - \Delta v} \tag{4.46}$$

where:

t = time during which through vehicle inspects gaps in the adjacent lane (sec).

The driver inspects the gap immediately after he/she notices the left-turning vehicle. The average number of short spaces inspected by the driver afterwards can be estimated using the moving-observer concept. The resulting equation to estimate the number of gaps inspected is as follows:

$$r = 1 + \frac{t \cdot v_t}{h_r} \cdot \left(1 - \frac{v_t - \Delta v}{v_t}\right) = 1 + \frac{t \cdot \Delta v}{h_r} \tag{4.47}$$

where:

h_r = average rejected space headway (m).

The average space headway shorter than critical can be estimated using the following equation:

$$h_r = \frac{\frac{v_t}{q_a} - exp\left(-\frac{h_2 \cdot q_a \cdot s - v_t \cdot q_a}{v_t \cdot (s - q_a)}\right) \cdot \left(h_2 + \frac{v_t}{q_a} - \frac{v_t}{s}\right)}{1 - exp\left(-\frac{h_2 \cdot q_a \cdot s - v_t \cdot q_a}{v_t \cdot (s - q_a)}\right)} \tag{4.48}$$

The model assumes that once the through vehicle is stopped behind the left-turning vehicle, the driver no longer attempts to change lanes.

4.4.3 Delay Caused by Left-Turn Maneuver

Figure 4.12 shows the vehicle arrival and departure curves when a left-turning vehicle blocks through traffic. The area between the curves gives the average delay caused by one left-turning vehicle. The initial flow rate in the shared lane before the left-turning vehicle is adjusted to reflect lane changes. Thus, the average delay caused by one left-turning vehicle is given by

$$d_t = \frac{q \cdot P_j \cdot s \cdot t_l^{\,2}}{2 \cdot \left(s - P_j \cdot q\right)} \tag{4.49}$$

where:

q = flow rate in shared lane before left-turning vehicle is observed (veh/sec/lane),

d_t = average delay caused by one left-turning vehicle (sec).

4.4.4 Sensitivity Analysis for Left-Turn Maneuver

Figures 4.13 to 4.15 show some sample results from a sensitivity analysis for the left-turn maneuver. The parameters used are listed in Table 4.3. An equal traffic distribution by lane has been assumed. The three graphs correspond to different values of P_{a2}. Within each graph, the values of blockage time and arterial volume vary. The results show that for a given blockage time, the average delay increases as the volume increases. As the arterial volume increases, more through vehicles are influenced by the left-turning vehicle. The average delay also increases as the blockage time of the left-turning vehicle increases.

A comparison between the graphs shows the effects of lane changing. As more drivers attempt to change lanes, the average delay caused by the left-turning vehicle decreases. The effect of lane changing becomes less significant at high arterial volumes due to the lack of availability of sufficient gaps in the adjacent lane.

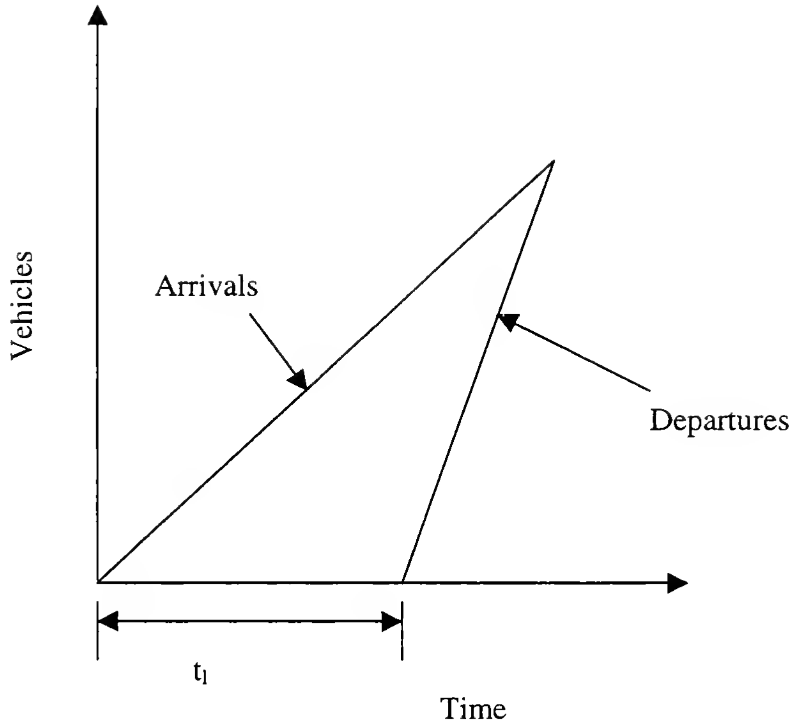

Figure 4.12 Vehicle arrival and departure curves for left-turn blockage

Table 4.3 Parameters used for sensitivity analysis for left-turn maneuver

Parameter	Value
S	0.5 veh/s
h_2	71.1 m
v_t	64 km/hr
Δv	15 km/hr
w_l	150 m

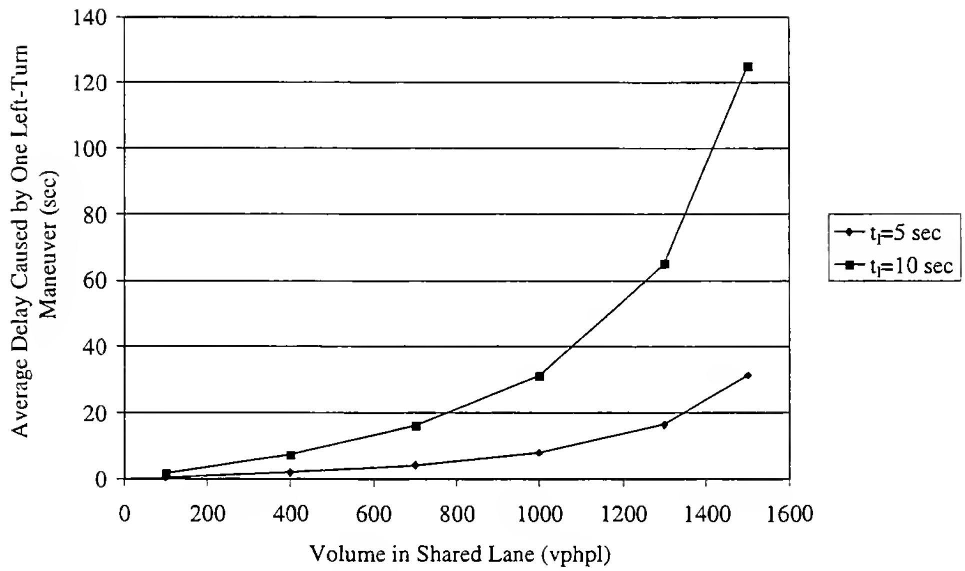

Figure 4.13 Delay caused by left-turn maneuver for P_{a2}=0

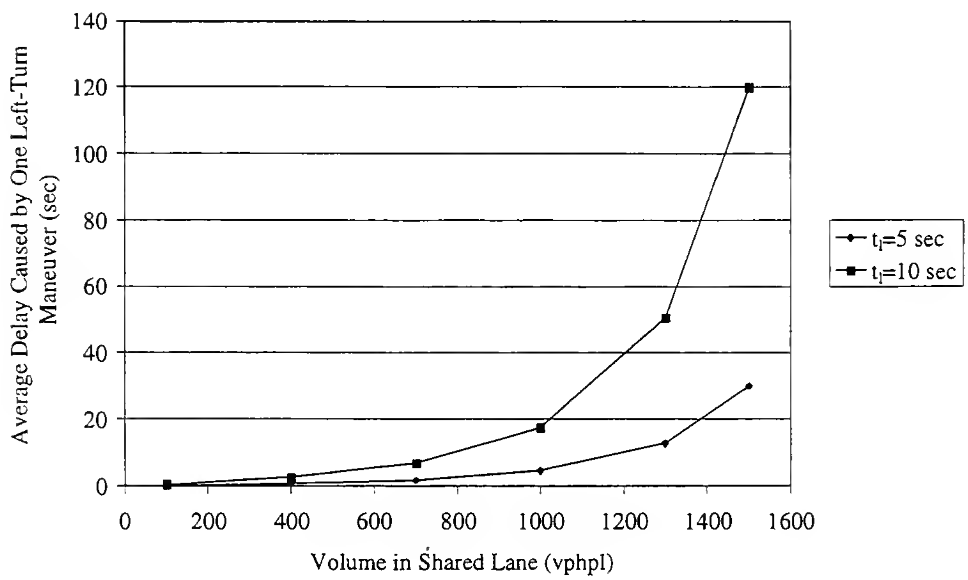

Figure 4.14 Delay caused by left-turn maneuver for P_{a2}=0.5

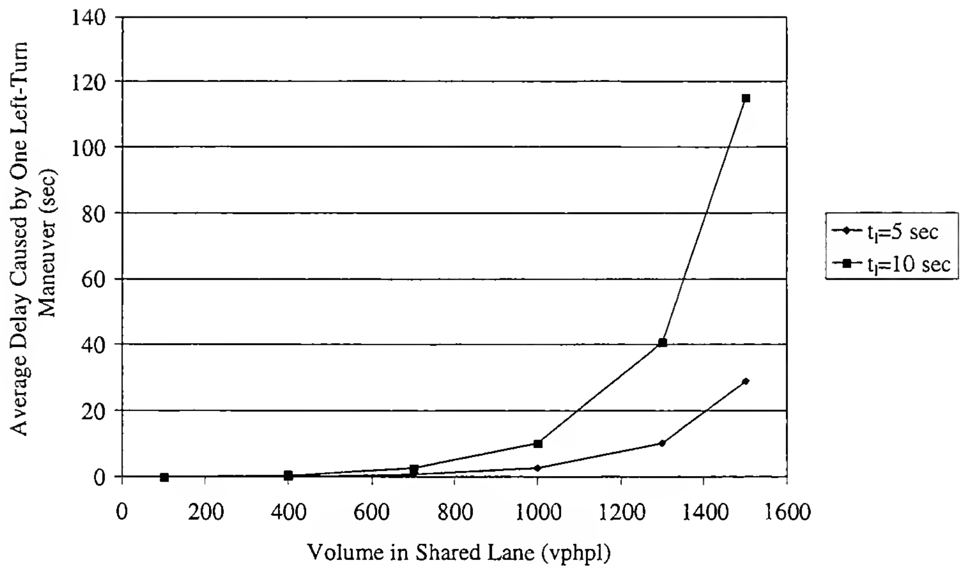

Figure 4.15 Delay caused by left-turn maneuver for P_{a2}=1

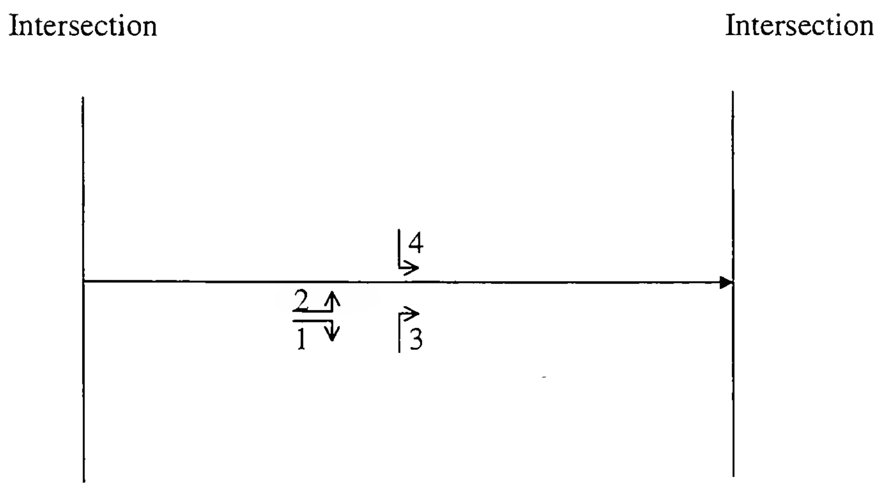

Figure 4.16 Traffic streams used by developed delay models

4.5 Use of Developed Delay Models

The delay models presented in this chapter can be used to estimate the delays between intersections. Each segment between intersections is analyzed separately in each direction since the two directions may have different traffic volumes. The input data that need to be provided by the user include movement volumes, arterial traffic distribution by lane, arterial speed, diverging distance, diverging speed, merging distance, merging speed, and the presence of an exclusive left-turn lane. Figure 4.16 shows the traffic streams incorporated into the analysis for one direction on a segment.

4.5.1 Effect of Minor Streams on Major Streams

The streams depicted in Figure 4.16 can cause delays to the arterial traffic. The delay models presented in this chapter can be used to estimate the arterial delays caused by minor streams. The impact of stream 1 can be estimated using the diverging model. The type of model used to estimate the delays caused by stream 2 varies based on whether an exclusive left-turn lane is present. For sections where exclusive left-turn lanes are not provided, the through and left-turning vehicles share a lane. In this case, the left-turning vehicle blocks the through vehicles, and the left-turn model presented in Section 4.4 is used to estimate the delay of the through vehicles. In the case of a section with exclusive left-turn lanes, it is assumed that sufficient storage space is provided so that left-turning vehicles do not block the through vehicles. However, the left-turning vehicle may still cause delay to the through vehicles as it decelerates to enter the left-turn lane. In this case, the diverging model should be applied to estimate the impact of the left-turning vehicle on the through vehicles. The diverging speed used in this case is the speed at which the vehicle exits the through lane and enters the left-turn lane. Streams 3 and 4 both represent cases where vehicles merge into the arterial traffic. For these streams, the merging model may be used to estimate the delays caused to through vehicles. Streams 3 and 4 may have different merging speeds and merging distances.

4.5.2 Delays of Minor Streams

The delays experienced by the minor streams due to the arterial streams can also be estimated. It is assumed in the calculations that the minor streams do not experience queuing. If there is an access point on the segment where minor streams experience queuing, the access point should be represented as an intersection in the network. Under the assumption of no queuing, the average delays of vehicles in streams 2, 3, and 4 are given by

$$d = \frac{3600}{c} \tag{4.50}$$

where:

d = average delay experienced by one vehicle in minor stream (sec/veh),

c = capacity calculated from *Highway Capacity Manual* (veh/hr).

The total delay experienced by a given minor stream can be calculated by multiplying the average delay per vehicle as calculated by Equation 4.50 by the number of minor stream maneuvers.

5. SAFETY ANALYSIS

One important benefit of access control is improved safety. A reduction in the number of individual access points or improvements to access points such as the addition of a two-way left-turn lane may help to reduce the number of crashes on the arterial. In order to evaluate the possible benefits of access control, impact models to predict crashes based on road geometric and access control characteristics need to be developed.

Safety analysis for the arterial consists of predicting crashes for different components of the network: intersections and segments. An important component of this study involved trying to develop regression models to predict crashes on urban multi-lane arterial segments in Indiana based on geometric and access control characteristics. In order to develop such a model, two major categories of data were needed: crash data for segments and data regarding segment geometric and access control characteristics.

5.1 Road Segment Data

The multi-lane road sections used as a sample to develop regression models were selected in cooperation with INDOT to represent a wide array of geographic locations and levels of access control. Table 5.1 summarizes the sections used for analysis. These sections were subdivided into segments defined in the INDOT Road Inventory Database (RIDB) for analysis. The RIDB segments are homogeneous with respect to cross section and traffic volume.

The RIDB classifies each segment based on access control into three qualitative levels. For the purpose of this study, more detailed access control data regarding the number and type of access points were needed. These data were obtained from the INDOT videolog database. The segments were viewed on the videolog database to obtain data regarding access points and cross section. Data were collected from the videolog database for approximately 150 segments over a two-week period.

Table 5.1 List of sections used in safety analysis

County	Route	Section
Allen	Old US 24/30	Goshen Av. to Anthony Blvd.
Bartholomew	SR 46	I-65 to SR 11
Clark	SR 131	SR 62 to I-65
Clark	SR 62	Conrail #308 to Springdale Dr.
Howard	US 31	Alto Rd. to Sycamore St.
Jefferson	SR 62	SR 7 to US 421
Lake	US 30	SR 55 to SR 53
Lake	US 30	Mississippi St. to SR 51
Lake	US 41	US 30 to I-80
Madison	SR 9	SR 236 to SR 32
Marion	SR 135	County Line Rd. to David-Lind Dr.
Marion	SR 37	US 31 to Fall Creek Pkwy.
Marion	US 31	Kessler Blvd. to 86th St.
Marion	US 31	County Line Rd. to Thompson Rd.
Marion	US 31	Mills Av. to Pleasant Run Pkwy.
Marion	US 36	Raceway Rd. to High School Rd.
Marion	US 40	I-465 to German Church Rd.
St. Joseph	US 33	I/80-90 connector to Glendale Av.
Tippecanoe	SR 26	US 52 to I-65
Tippecanoe	US 52	US 231 to SR 26
Vanderburgh	SR 66	US 41 to I-164
Vanderburgh	US 41	SR 66 to SR 57
Vigo	US 41	I-70 to US 40

5.2 Crash Data

The crash data for the segments were obtained from the INDOT Crash Database. In order to obtain the crash data, a software package had to be developed because the method for coding crash location in the INDOT Crash Database is not compatible with the RIDB. In the Crash Database, crash location is determined by a distance and direction from a reference point. The reference point is defined by a main street pseudo number and cross street pseudo number. In the RIDB, segments are arranged based on geographic order in each county. The RIDB does not contain pseudo number information.

5.2.1 County Files

Due to the incompatibility of the Crash Database and RIDB, an interface between these two databases is needed. Recent research at Purdue (Weiss, 1996) led to the development of RIDB matching software to attach pseudo numbers to the RIDB files. The resulting files with pseudo numbers attached, which hereafter are referred to as the County Files, provide an interface between the crash data and the RIDB data and are thus an important input to crash extraction software.

After running the RIDB matching software, further processing of the County Files was needed before crash extraction software could be run. The geographic direction in which the sections are listed was added manually. The information regarding the geographic direction of the routes was obtained from maps. In addition, main street pseudo numbers corresponding to local names of US and State roads were added to the County Files. These additional pseudo numbers are especially important in urban areas where it may be common practice for police to use local street names instead of route numbers when coding crash locations. Information regarding local street names was not consistently available in the RIDB and was therefore obtained from street maps. Some cross street pseudo numbers that could not be automatically processed by the RIDB matching software due to inconsistencies were added to the County Files. Inconsistencies in road names between the RIDB file and pseudo number list file may result from misspellings, use of incorrect suffixes such as Dr. instead of Ln., or abbreviated street

names. Even with manual processing, some cross street pseudo numbers could not be matched because the source of the discrepancy between the RIDB and pseudo number files could not be determined. The County Files were run through a Microsoft Excel macro developed by Eranky et al. (1997). The macro formats the County Files and calculates the segment lengths.

5.2.2 Crash Database File

The Crash Database File is also used as input to the crash extraction software. The Crash Database File is a text file and is based on the environmental records with information pertaining to crash location and severity. The information extracted includes county number, severity, main street pseudo number, cross street pseudo number, direction, and distance. Each Crash Database File corresponds to one year of crash data in Indiana.

5.2.3 Algorithm to Extract Crashes

In order to obtain the crash data, an algorithm was developed to extract crashes from the Crash Database and locate them on RIDB segments. Computer software was then developed based on the algorithm. The computer program was written in C++ by Nakarin Satthamnuwong, a graduate student at Purdue. The crash extraction software uses the County File and the Crash Database File as input. The software extracts crashes for a given county and given year during each run. The county number being run, name of the County File, and name of the Crash Database File are passed to the program as arguments in the command line.

Figure 5.1 shows the general algorithm for the software to extract crashes. The software first screens the Crash Database File for crash records with a matching county number, complete data for crash location, and a unique matching pseudo number pair in the County File. Screening for complete data is needed because many records in the Crash Database File have missing pseudo numbers, distance, or direction. These crashes

cannot be located. After checking for missing data, the County File is checked to see if a unique matching pseudo number pair can be found. In many cases, a matching pair in the County File cannot be found, mainly because the County File only contains Interstate, US, and State roads. In other cases, more than one matching pseudo number pair may be found in the County File because the same pseudo number pair can represent more than one physical location. These crashes cannot be located. When a unique matching pair has been found in the County File, the record containing this pair in the County File corresponds to the reference intersection from which the direction and distance were measured. Records in the county being run with complete data and a unique matching pair are saved in a temporary file called *temp1.txt* for further use.

Once the Crash Database File has been screened, the crash direction of each crash record in *temp1.txt* is read and compared with the crash direction in the County File to determine whether to move up or down in the County File to locate the crash. In some cases, the direction to search the County File cannot be determined, and the crash cannot be located. The search direction for each record is saved with other information regarding crash location in another temporary file called *temp2.txt*. Each record in *temp2.txt* is then read, and segment lengths in the County File are added until the crash is located. In some cases, a crash may be missed because the distance coded in the Crash Database File is greater than the remaining distance on the route in the County File. If the crash distance is valid and the segment of the crash is located, the distance to the segment endpoints must be checked to determine if the crash occurred near a segment endpoint. Crashes that occur within 30 m of the segment endpoints are not counted because they are associated with intersections or changes in cross section. For crashes occurring on the interior of a segment, the severity is checked, and the crash statistics for the segment are updated.

The program produces two output files in text format. The first file is a table containing a list of unique DRK numbers that define each segment. For each segment, the following information is given: total crashes, fatal crashes, injury crashes, property-damage-only (pdo) crashes, and number of crashes for which severity was not coded in the Crash Database File. The second file contains summary statistics for the program run, including the total number of crashes in Indiana for a given year, the number of crashes

in the county that was run, the number of crashes that were missed for various reasons, the number of crashes located near the segment endpoints, and the number of crashes located on segments. The information regarding the number of crashes missed was used to develop an adjustment factor to account for missing crashes.

5.2.4 Obtaining Crash Data

Once the software to extract crashes was developed and tested, the software could be used to obtain the crash data for the segments in this study. The crash extraction software was run for 12 counties and 5 years. The counties for which crashes were extracted are as follows:

- Allen
- Bartholomew
- Clark
- Howard
- Jefferson
- Lake
- Madison
- Marion
- St. Joseph
- Tippecanoe
- Vanderburgh
- Vigo.

For each county, crash data from 1991 to 1995 were obtained. The output files from the crash extraction software were used to obtain the crash data for the segments in this study. In most cases, the total number of crashes over five years was used. For a few segments that underwent improvements during the period from 1991 to 1995, the total number of crashes over three years was used to ensure that the segments had consistent cross section characteristics.

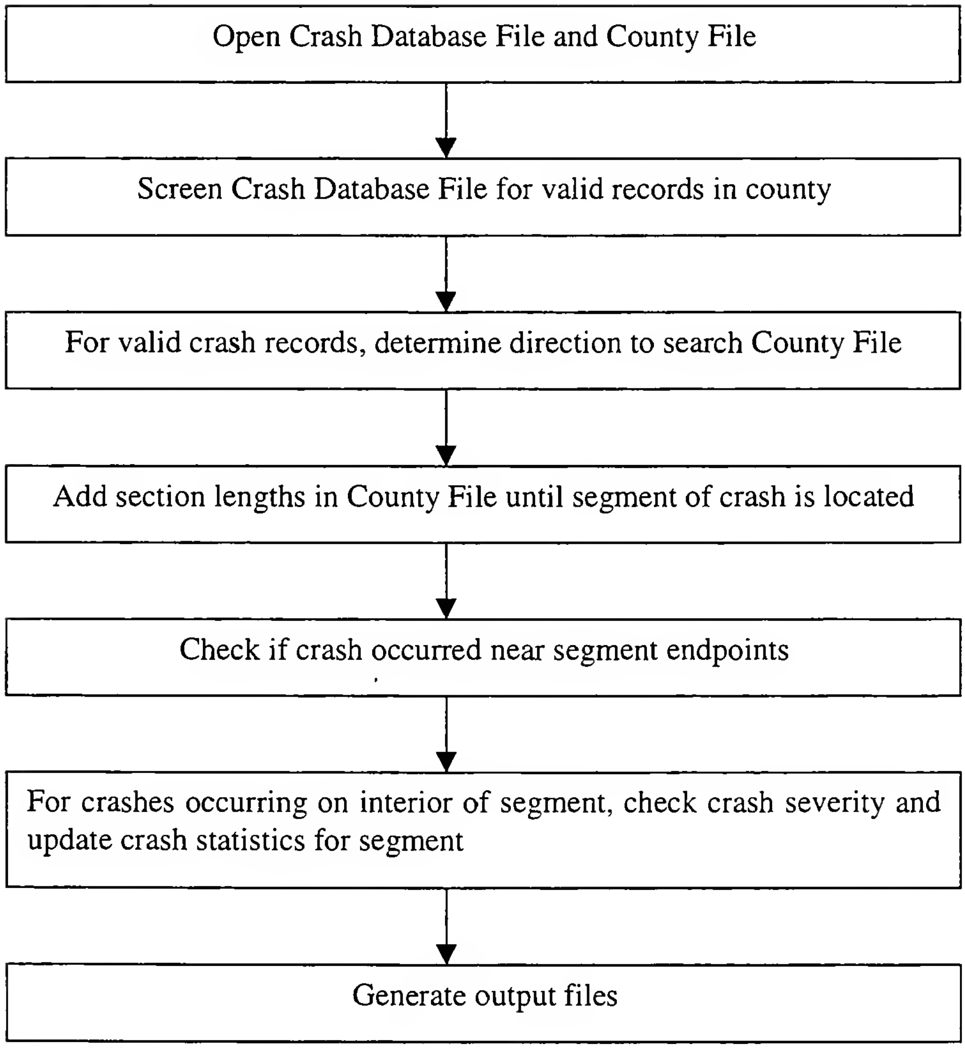

Figure 5.1 Summary of algorithm to extract crashes

5.3 Statistical Models

Once the crash and segment data were collected, statistical models were developed to predict the number of total crashes, number of fatal and injury crashes, and number of pdo crashes. A negative binomial regression model was used with the following form:

$$Y = k \cdot LEN \cdot YRS \cdot AADT^{\gamma} \cdot exp\left(\sum_i (\beta_i \cdot X_i)\right) \tag{5.1}$$

where:

Y = number of total, fatal/injury, or pdo crashes,

k = intercept coefficient,

LEN = length of the segment,

YRS = number of years,

$AADT$ = Annual Average Daily Traffic,

γ, β_i = model parameters,

X_i = variables representing segment characteristics.

This model was recently utilized by Eranky et al. (1997) in a study to develop crash reduction factors for Indiana segments.

5.3.1 Development of Statistical Models

The statistical analysis was done using LIMDEP v7.0. The following is a list of variable names with a brief description and units where applicable:

CRASH Total number of crashes on the segment.

PDO Number of pdo crashes on the segment.

FATINJ Number of fatal and injury crashes on the segment.

LEN Length of the segment (km).

YRS Number of years.

AADT	Annual Average Daily Traffic (thousands of vehicles).
ACCESS	Access density (per km).
SHLDR	Dummy variable to indicate presence of outside shoulder (1 if outside shoulder is present, 0 otherwise).
SL	Speed limit on the segment (km/hr).
LANES	Number of lanes on the segment.
COMM	Dummy variable to indicate segment in commercial area (1 if area type is commercial, 0 otherwise).
PS	Proportion of access points that are signalized.
TWLTL	Dummy variable to indicate presence of two-way left-turn lane on segment (1 if two-way left-turn lane is present, 0 otherwise).
NOMED	Dummy variable to indicate segment without median or two-way left-turn lane (1 if segment has no median, 0 otherwise).
NOMEDO	Dummy variable to indicate segment with median (excluding two-way left-turn lane) with no openings between signalized intersections (1 if segment has median with no openings between signals, 0 otherwise).
PC	Proportion of access points that are channelized.
PR	Proportion of access points with right-turn lane.
LNLEN	Natural logarithm of *LEN*.
LNYRS	Natural logarithm of *YRS*.
LNAADT	Natural logarithm of *AADT*.

The AADT data were obtained from the RIDB. The segment lengths were calculated from the RIDB using the Excel macro developed by Eranky et al. (1997). The segment lengths as calculated by the macro were adjusted by subtracting approximately 0.06 km because crashes near the segment endpoints are associated with intersections or changes in cross section. The natural logarithms of *LEN*, *YRS*, and *AADT* were used as input to LIMDEP v7.0 to transform the model to the form of Equation 5.1.

The access density was calculated as the total number of access points divided by the segment length. The total number of access points includes both signalized and

unsignalized access points. For unsignalized intersections, a T-intersection was considered as one access point, while an all-way intersection was considered as two access points. Signalized intersections were considered as two access points since traffic may have to stop at the signal in either direction on the segment. Access points within 30 m of the segment endpoints were not considered. The proportion of signalized access points was calculated as the number of signalized access points divided by the total number of access points. This value was defined to be zero for a segment with no access points. The proportion of channelized access points was calculated as the number of channelized access points divided by the total number of access points. The proportion of access points with a right-turn lane was calculated as the number of access points with a right-turn lane divided by the total number of access points.

Separate models were developed to predict the total number of crashes, number of pdo crashes, and number of fatal/injury crashes. The development of the three models followed a similar process. A significance level of 0.10 was used. First, a basic model was run with the basic cross section variables *ACCESS*, *SL*, *COMM*, *LANES*, and *SHLDR*. Access density appeared to have the greatest influence on crash occurrence. Then four separate models were run using two variables: *ACCESS* and one of *SL*, *COMM*, *LANES*, and *SHLDR*. The combination of *ACCESS* and *SHLDR* was significant. To the initial model containing *ACCESS* and *SHLDR*, the median variables and *PS* were added. The *PS* variable was significant, while all of the median variables with the exception of *NOMED* were significant.

The variables *PC* and *PR* were then tested to determine if the type of access point had an impact on crashes. The results for these models for the dependent variable *CRASH* are shown in Tables 5.2 to 5.4. The variables *PC* and *PR* were individually significant but were not significant when both were introduced into the model simultaneously. In addition, the signs of the coefficients for *PC* and *PR* were positive. This result seems to indicate that channelization and right-turn lanes increase crashes. However, channelization and right-turn lanes are more likely to be installed at high-volume access points. When *PC* and *PR* are large, the proportion of high-volume access points would also typically be large. More crashes would be expected due to high-volume access points

than low-volume access points. Thus, the model results for *PC* and *PR* seem to indicate the effect of high-volume access points. Because volume at access points was outside the scope of data that could be collected from the videolog database, the effects of volume at access points, channelization, and right-turn lanes cannot be separated. Therefore, it was decided not to use the models incorporating channelization and right-turn lanes.

Table 5.2 Regression model including PC variable (dependent variable = CRASH)

Variable	Coefficient	Standard Error	b/St.Er.	P[|Z|>z]	Mean of X
Constant	-1.321	0.291	-4.543	0.0000	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0295	0.00784	3.758	0.0002	22.644
SHLDR	-0.601	0.241	-2.492	0.0127	0.481
PS	2.022	0.903	2.240	0.0251	0.0804
TWLTL	-0.734	0.315	-2.327	0.0200	0.150
NOMEDO	-0.712	0.267	-2.671	0.0076	0.293
PC	1.548	0.773	2.002	0.0453	0.0921
Overdispersion parameter for negative binomial model					
Alpha	1.098	0.133	8.259	0.0000	

Table 5.3 Regression model including PR variable (dependent variable = CRASH)

Variable	Coefficient	Standard Error	b/St.Er.	P[|Z|>z]	Mean of X
Constant	-1.366	0.297	-4.602	0.0000	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0306	0.00824	3.712	0.0002	22.644
SHLDR	-0.631	0.254	-2.487	0.0129	0.481
PS	2.423	0.869	2.788	0.0053	0.0804
TWLTL	-0.667	0.328	-2.033	0.0421	0.150
NOMEDO	-0.666	0.246	-2.701	0.0069	0.293
PR	0.525	0.258	2.033	0.0421	0.254
Overdispersion parameter for negative binomial model					
Alpha	1.123	0.128	8.781	0.0000	

Table 5.4 Regression model including PC and PR variables (dependent variable = CRASH)

Variable	Coefficient	Standard Error	b/St.Er.	P[IZI>z]	Mean of X
Constant	-1.401	0.290	-4.824	0.0000	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0304	0.00790	3.849	0.0001	22.644
SHLDR	-0.616	0.242	-2.547	0.0109	0.481
PS	2.019	0.850	2.376	0.0175	0.0804
TWLTL	-0.689	0.311	-2.220	0.0264	0.150
NOMEDO	-0.734	0.258	-2.847	0.0044	0.293
PC	1.318	0.873	1.510	0.1310	0.0921
PR	0.318	0.272	1.169	0.2422	0.254
Overdispersion parameter for negative binomial model					
Alpha	1.089	0.131	8.340	0.0000	

Table 5.5 Regression model for total number of crashes (before adjustment for missing crashes)

Variable	Coefficient	Standard Error	b/St.Er.	P[IZI>z]	Mean of X
Constant	-1.182	0.298	-3.972	0.0001	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0285	0.00843	3.379	0.0007	22.644
SHLDR	-0.631	0.257	-2.455	0.0141	0.481
PS	2.520	0.913	2.761	0.0058	0.0804
TWLTL	-0.748	0.343	-2.179	0.0293	0.150
NOMEDO	-0.604	0.228	-2.649	0.0081	0.293
Overdispersion parameter for negative binomial model					
Alpha	1.147	0.131	8.780	0.0000	

Thus, the final models contained the following variables: *ACCESS*, *SHLDR*, *PS*, *TWLTL*, *NOMEDO*. Based on the model results, the parameter γ did not appear to differ significantly from one. It was decided to fix this parameter at one. Models with unrestricted coefficients for *LEN* and *YRS* were also tested; these coefficients did not appear to differ significantly from one. All three models had a similar structure. Tables 5.5 to 5.7 provide a summary of the regression output before adjustment for missing crashes.

Table 5.6 Regression model for property-damage-only crashes (before adjustment for missing crashes)

Variable	Coefficient	Standard Error	b/St.Er.	P[IZI>z]	Mean of X
Constant	-1.459	0.277	-5.272	0.0000	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0261	0.00814	3.210	0.0013	22.644
SHLDR	-0.669	0.250	-2.680	0.0074	0.481
PS	2.627	0.861	3.050	0.0023	0.0804
TWLTL	-0.686	0.321	-2.134	0.0329	0.150
NOMEDO	-0.684	0.212	-3.234	0.0012	0.293
Overdispersion parameter for negative binomial model					
Alpha	1.105	0.134	8.260	0.0000	

Table 5.7 Regression model for fatal/injury crashes (before adjustment for missing crashes)

Variable	Coefficient	Standard Error	b/St.Er.	P[IZI>z]	Mean of X
Constant	-2.540	0.312	-8.141	0.0000	
LNLEN	1	........(Fixed Parameter)........			-0.538
LNYRS	1	........(Fixed Parameter)........			1.583
LNAADT	1	........(Fixed Parameter)........			3.431
ACCESS	0.0325	0.00779	4.166	0.0000	22.644
SHLDR	-0.525	0.252	-2.081	0.0374	0.481
PS	2.280	0.870	2.620	0.0088	0.0804
TWLTL	-0.865	0.356	-2.432	0.0150	0.150
NOMEDO	-0.493	0.252	-1.956	0.0505	0.293
Overdispersion parameter for negative binomial model					
Alpha	1.041	0.138	7.540	0.0000	

Residual plots and plots of the observed versus predicted values for the three models before adjustment for missing crashes are given in Figures 5.2 to 5.7. The plots of the observed versus predicted values appear to be symmetric about the 45 degree line. The residual plots indicate that the variance appears to be increasing as the predicted values increase.

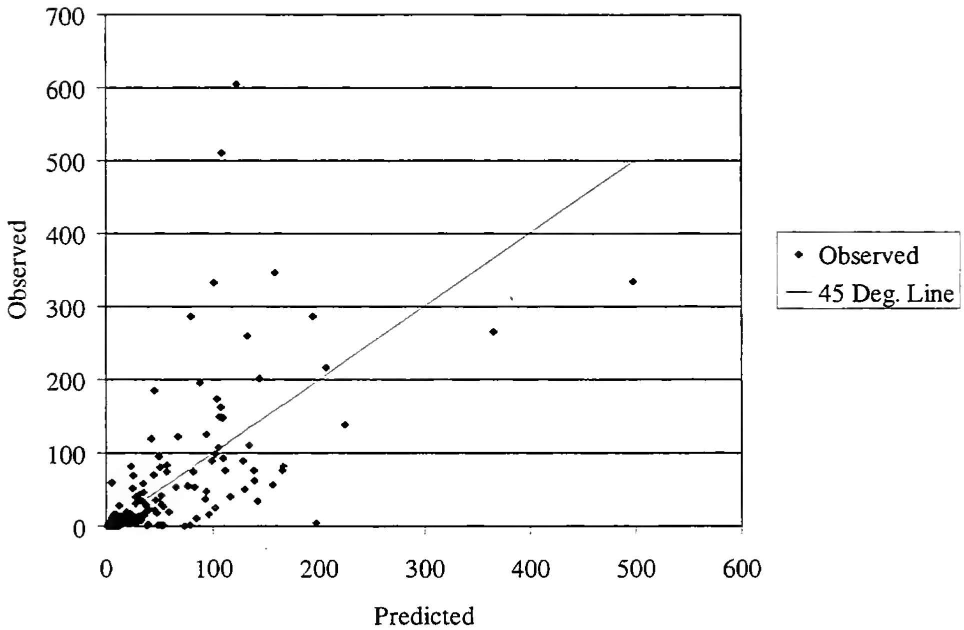

Figure 5.2 Observed versus predicted for dependent variable CRASH

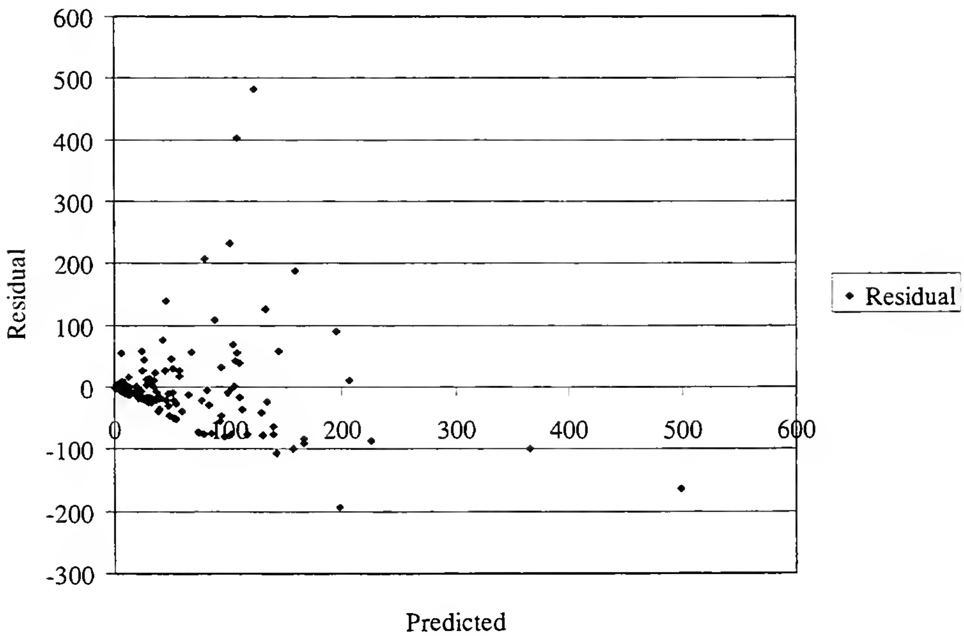

Figure 5.3 Residual plot for dependent variable CRASH

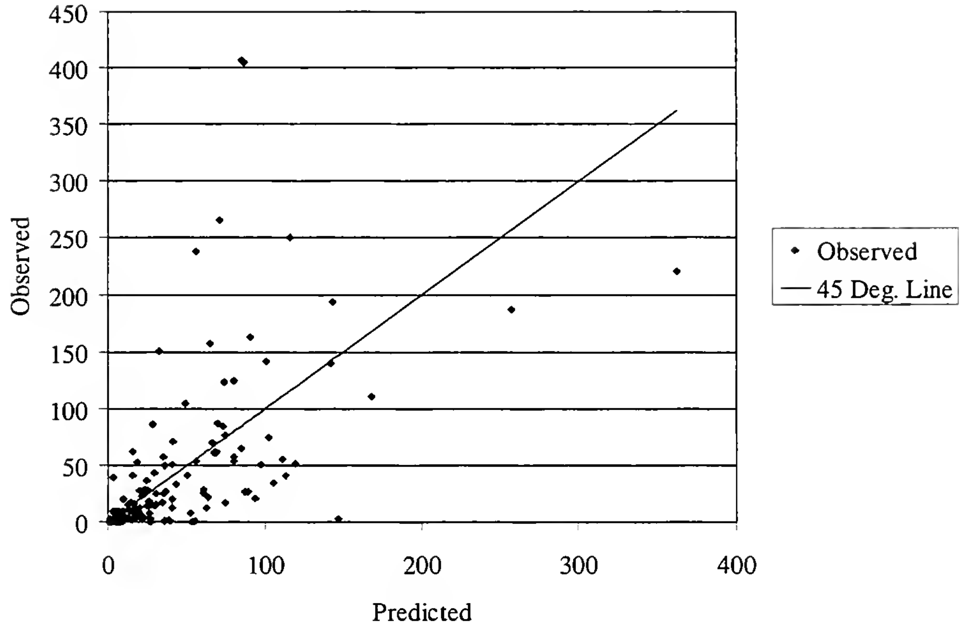

Figure 5.4 Observed versus predicted for dependent variable PDO

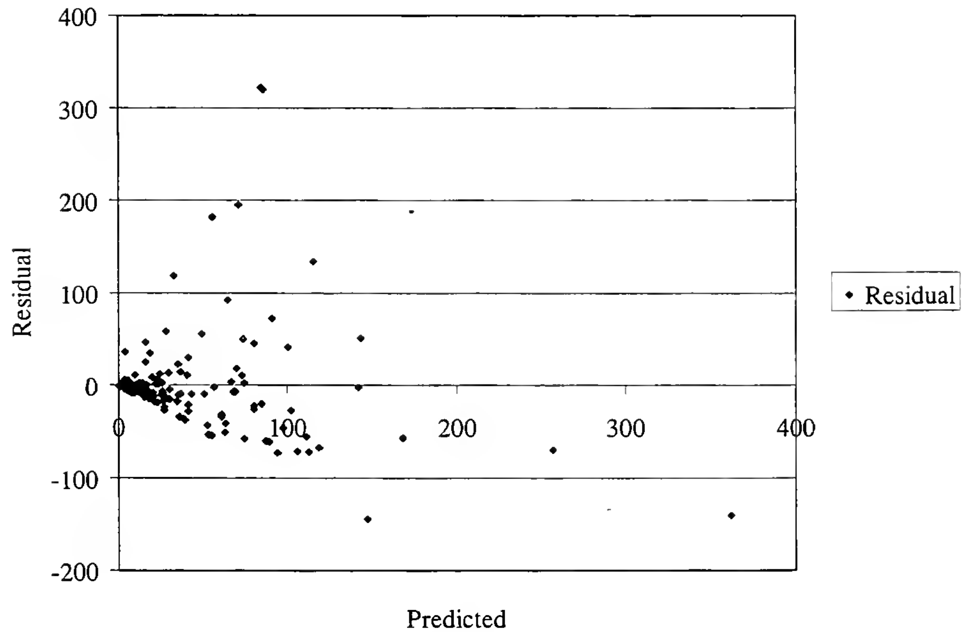

Figure 5.5 Residual plot for dependent variable PDO

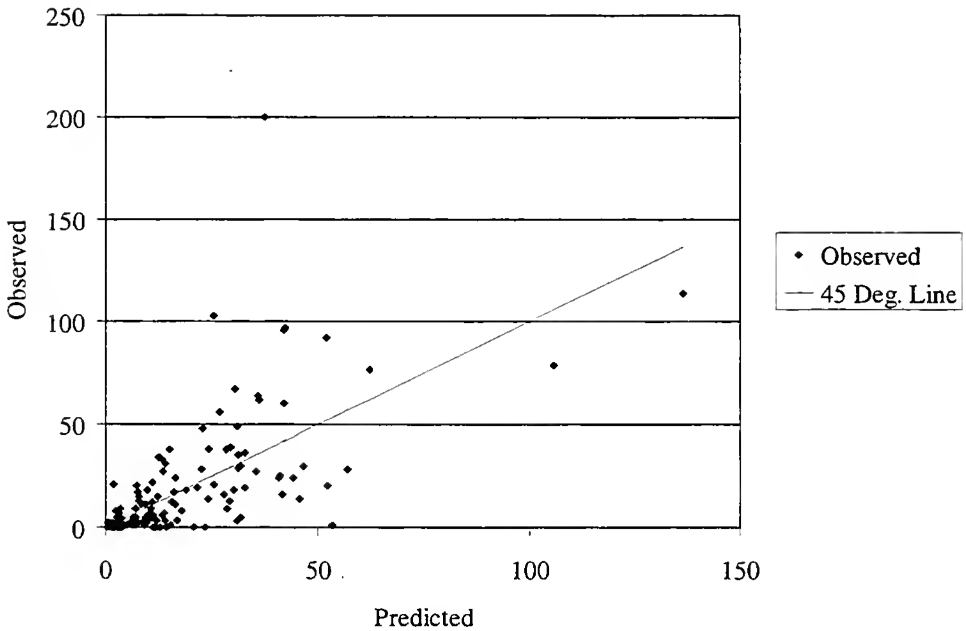

Figure 5.6 Observed versus predicted for dependent variable FATINJ

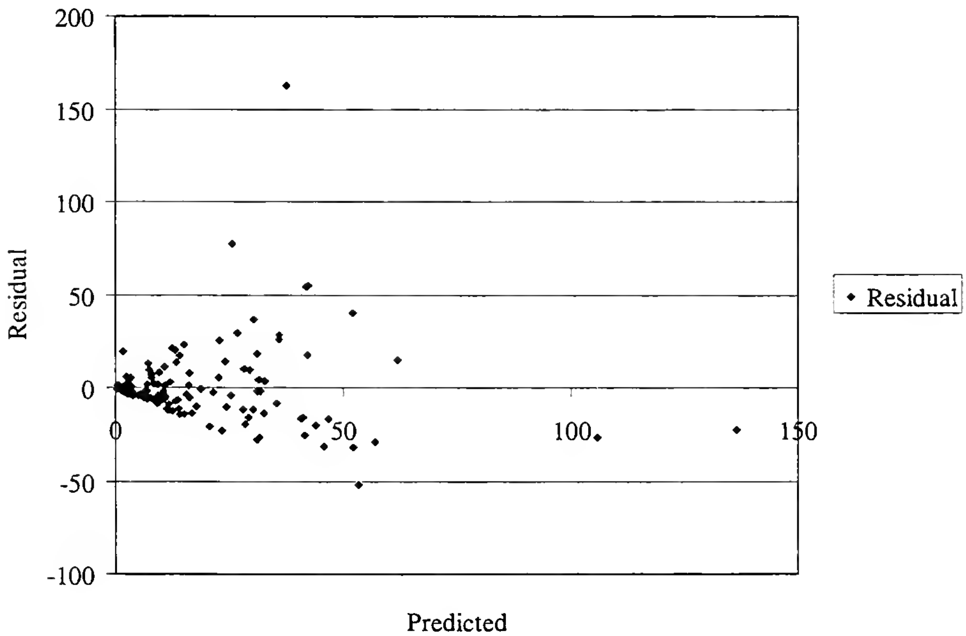

Figure 5.7 Residual plot for dependent variable FATINJ

5.3.2 Adjustment for Missing Crashes

The models were then adjusted to account for missing crashes. The summary statistics generated from the crash extraction software were used to determine the proportion of missing crashes. Crashes could be missed for several reasons. First, a crash record may contain missing pseudo numbers, distance, or direction. Crashes may also be missed if the pseudo number pair in the crash record corresponds to more than one physical location in the County File. Finally, the crash direction or distance may be coded incorrectly. It was assumed that the loss of a given crash under given circumstances could happen to any crash in the Crash Database with the same likelihood. Thus, a single adjustment factor was developed to account for missing crashes. The adjustment factor was determined based on the 12 counties in this study and 5 years of crash data. Table 5.8 summarizes the proportion of crashes lost. Based on these results, an adjustment factor of 1.61 ($=1/((1-0.311)*(1-0.083)*(1-0.018))$) was applied to account for missing crashes.

Table 5.8 Percent of crashes lost

Reason missed	Percent missed
Missing data	31.1
Multiple locations	8.3
Incorrect distance or direction	1.8

5.3.3 Final Equations

The final equations after adjustment for missing crashes are as follows:

$$\begin{aligned} CRASH = 0.494 \cdot LEN \cdot YRS \cdot AADT \cdot exp(& 0.0285 \cdot ACCESS - 0.631 \cdot SHLDR \\ & + 2.520 \cdot PS - 0.748 \cdot TWLTL - 0.604 \cdot NOMEDO) \end{aligned} \tag{5.2}$$

$$\begin{aligned} PDO = 0.374 \cdot LEN \cdot YRS \cdot AADT \cdot exp(& 0.0261 \cdot ACCESS - 0.669 \cdot SHLDR \\ & + 2.627 \cdot PS - 0.686 \cdot TWLTL - 0.684 \cdot NOMEDO) \end{aligned} \tag{5.3}$$

$$FATINJ = 0.127 \cdot LEN \cdot YRS \cdot AADT \cdot exp(0.0325 \cdot ACCESS - 0.525 \cdot SHLDR + 2.280 \cdot PS - 0.865 \cdot TWLTL - 0.493 \cdot NOMEDO) \tag{5.4}$$

The units of the variables in Equations 5.2 to 5.4 are as follows:

LEN (km),

AADT (thousands of vehicles),

ACCESS (km^{-1}).

5.4 Discussion of Results

All three of the models have a similar structure. The following variables were found to have a significant effect on crash occurrence: *ACCESS*, *SHLDR*, *PS*, *TWLTL*, and *NOMEDO*.

The coefficient of *ACCESS* is positive, indicating that segments with more frequent access points experience more crashes. This result is expected since the introduction of more access points creates more conflict points between through and minor stream vehicles. The coefficient of *SHLDR* is negative, indicating that the presence of an outside shoulder leads to a reduction in crashes. An outside shoulder may increase drivers' comfort by increasing the traveled way width. In addition, the introduction of an outside shoulder may lead to increased turning radii and thus higher merging and diverging speeds. The coefficient of *PS* is positive, indicating that the presence of signals can lead to higher crash rates. This result may be due to the higher likelihood of rear-end collisions when vehicles are stopped at signals. The presence of a two-way left-turn lane leads to a reduction in crashes. The presence of a median with no openings between signalized intersections also leads to a reduction in crashes. In this case, there are no left-turn or crossing maneuvers permitted at unsignalized access points.

5.5 Results from Prior Study

The results from Eranky et al. (1997) may be used to estimate the number of crashes for multi-lane segments in the impact area but not on the arterial and for two-lane segments. The basic calibrated models include *AADT* (thousands of vehicles) and *LEN* (km) as the predictor variables. The following are the results from Eranky et al. (1997):

Urban Multi-Lane:

$$CRASH = 0.11602 \cdot LEN \cdot AADT^{1.3304} \tag{5.5}$$

$$PDO = 0.23031 \cdot LEN \cdot AADT^{1.1009} \tag{5.6}$$

$$FATINJ = 0.0076528 \cdot LEN \cdot AADT^{1.665} \tag{5.7}$$

Urban Two-Lane:

$$CRASH = 0.16550 \cdot LEN \cdot AADT^{1.3287} \tag{5.8}$$

$$PDO = 0.088600 \cdot LEN \cdot AADT^{1.415} \tag{5.9}$$

$$FATINJ = 0.062552 \cdot LEN \cdot AADT^{1.2236} \tag{5.10}$$

Equations 5.5 to 5.10 predict the number of crashes per year.

5.6 Crash Rates at Signalized Intersections

In addition to crash rates for segments, crash rates for signalized intersections can also be estimated for signalized intersections that are located at the endpoints of arterial segments or located on segments outside the arterial but in the impact area. Regression

models to predict crash rates at signalized intersections were recently developed by Jonathan Weiss, a graduate student at Purdue University. These results have not yet been published. The equations Jonathan Weiss developed to predict the annual number of crashes are as follows:

$$PDO = 1.1050 \times 10^{-4} \cdot APPVOL1^{0.65010} \cdot APPVOL2^{0.52946} \cdot exp(-0.37831 \cdot LFTPROB - 0.11651 \cdot NUMDIV) \tag{5.11}$$

$$FATINJ = 4.4571 \times 10^{-4} \cdot APPVOL1^{0.38638} \cdot APPVOL2^{0.37357} \cdot exp(0.36144 \cdot NUMAPP) \tag{5.12}$$

where:

PDO = number of pdo crashes at the signalized intersection in representative year,

FATINJ = number of fatal and injury crashes at the signalized intersection in representative year,

APPVOL1 = average volume on N-S approaches (vehicles),

APPVOL2 = average volume on E-W approaches (vehicles),

NUMAPP = number of intersection approaches (2, 3, or 4),

LFTPROB = number of left-turning movements forbidden on all approaches (0, 1, 2, 3, or 4),

NUMDIV = number of approaches on which traffic is divided by median.

5.7 Use of Safety Models

The crash models presented in this chapter can be used to estimate the number of crashes for segments between intersections. The segments should be homogeneous with respect to cross section, AADT, and level of access control. The total number of crashes, fatal/injury crashes, and pdo crashes can be estimated from the models. Equations 5.2 to

5.4 can be applied to the multi-lane arterial segments. Equations 5.5 to 5.7, which are from the study by Eranky et al. (1997), can be applied to urban multi-lane segments in the impact area but not on the arterial. Equations 5.8 to 5.10, which are also from the study by Eranky et al. (1997), can be applied to urban two-lane segments in the impact area. Crash rates for signalized intersections located at the endpoints of arterial segments or outside the arterial can be estimated using Equations 5.11 and 5.12. Once the various models have been used for a given alternative, the results can be combined to find the total number of crashes per year by severity type under a given access control alternative. The crashes can be then converted to crash costs for each alternative.

The models are not intended to be used as an optimizing tool. Signals should still be installed where warranted. Each access control alternative should be evaluated separately for safety. Once each alternative has been evaluated for safety, the crash costs can be combined with the operating costs and agency costs to determine the economic effectiveness of each alternative.

6. ECONOMIC EVALUATION OF ALTERNATIVES

Once traffic delays and crash rates have been predicted for a given alternative, the economic effectiveness of the alternative can be estimated. The analysis of the economic effectiveness incorporates agency costs and user costs over the project lifetime. The user of the method has the flexibility to select the project lifetime and the discount rate. Figure 6.1 shows an example of a cash flow diagram. The agency costs include capital cost and continuing costs, while the user costs involve continuing costs.

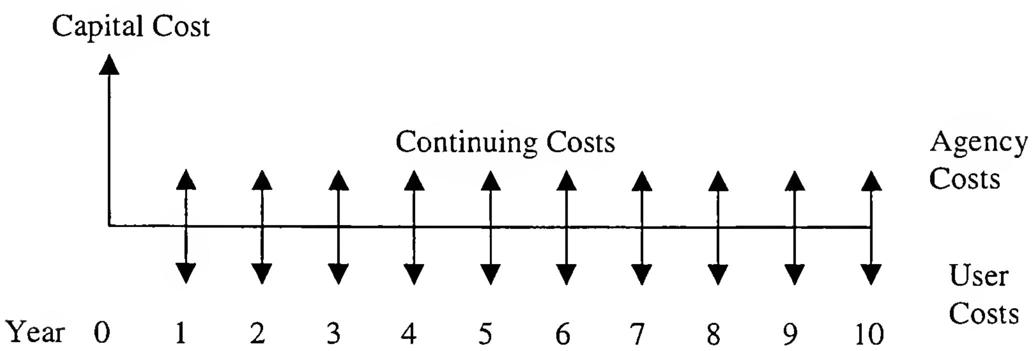

Figure 6.1 Example cash flow diagram

6.1 Agency Costs

The agency costs for each access control alternative are included in the analysis. The agency costs include capital and continuing costs. The capital cost consists primarily of the construction cost of a given alternative. The construction cost of each alternative can be estimated and incorporated into the analysis. The continuing costs involve the

annual maintenance costs of each alternative. These costs can be estimated and incorporated into the analysis. These costs can be converted to present worth and combined with the construction cost to obtain the present worth of the agency costs.

6.2 User Costs

The user costs of the project need to be estimated. The user costs include the operating costs and the crash costs. The project lifetime is represented by representative periods.

6.2.1 Selection of Representative Periods

The project lifetime is represented by typical periods. The user cost is predicted for the lifetime of each alternative project. During this period (10-30 years) the following traffic variations are expected:

- long-term but relatively slow changes caused by the region-wide trends and local gradual land development,
- abrupt changes caused by the appearance of strong local traffic generators,
- seasonal changes in an annual cycle,
- day-to-day changes in a weekly cycle,
- short-term changes in a daily cycle.

The abrupt changes in traffic pattern determine periods with slow traffic growth. Each such sub-period can be represented by a mid-year as shown in Figure 6.1. Alternatively, the first and final years of the sub-period can be used as representative years, and costs between these years can be estimated using linear interpolation. Each representative year is represented by a typical weekday. Where it is justified and possible, also a weekend day can be considered.

6.2.2 Estimation of Operating Costs

For delay analysis, a typical day is divided into several intervals within which traffic volumes are assumed to have a linear effect on delays. Each interval is represented by an average hour. Thus, a typical day is represented by several typical hours. For each hour, delays are estimated. The delays for each representative hour can be converted to fuel consumption and hourly operating costs using the TRANSYT-7F models. The following equations to estimate fuel consumption and operating costs are from Wallace et al. (1991b):

$$F = K_{1FC} \cdot TT + K_{2FC} \cdot D + K_{3FC} \cdot S \tag{6.1}$$

$$K_{1FC} = A_{11FC} + A_{12FC} \cdot V + A_{13FC} \cdot V^2 \tag{6.2}$$

$$K_{2FC} = A_{21FC} + A_{22FC} \cdot V + A_{23FC} \cdot V^2 \tag{6.3}$$

$$K_{3FC} = A_{31FC} + A_{32FC} \cdot V + A_{33FC} \cdot V^2 \tag{6.4}$$

$$C = [(K_{1OC} \cdot TT + K_{2OC} \cdot S + DC \cdot D)/1000 + FC \cdot F + O \cdot TC \cdot (TT/V + D)] \cdot I \tag{6.5}$$

$$K_{1OC} = A_{11OC} + A_{12OC} \cdot V + A_{13OC} \cdot V^2 + A_{14OC} \cdot V^3 \tag{6.6}$$

$$K_{2OC} = A_{21OC} + A_{22OC} \cdot V + A_{23OC} \cdot V^2 + A_{24OC} \cdot V^3 \tag{6.7}$$

where:

F = fuel consumption (lit),

TT = total travel (veh-km),

D = total delay (veh-hr),

S = total stops (vph),

V = cruise speed (km/hr),

A_{ijFC} = model coefficients for fuel consumption model,

DC = unit cost of vehicle delay (\$/1000 veh-hr),

FC = cost of fuel consumption (\$/lit),

O = vehicle occupancy (persons/veh),

TC = unit cost of passenger time (\$/pers-hr),

I = inflation factor to convert costs from base year of 1987,

A_{ijOC} = model coefficients for operating cost model.

Wallace et al. (1991b) also provide default values for model parameters and unit costs. The default values after conversion to metric units are listed in Table 6.1. The unit cost values are in 1987 dollars. The user should select the appropriate inflation factor I to convert the operating costs to the chosen base year. Other values for the other parameters could be specified by the user if needed.

Once the delays for the representative hours are converted to hourly operating costs, the hourly operating costs can be converted to daily operating costs. The daily operating costs can be converted to annual operating costs. Once the annual operating costs are estimated for the representative years, the operating costs can be converted to present worth for the entire lifetime.

6.2.3 Estimation of Crash Costs

To estimate crash costs, crash rates first need to be calculated for the impact area. Once crashes are predicted by severity, they can be converted to crash costs by using unit values for the cost of fatal/injury and pdo crashes. The crash costs can then be converted to present worth to obtain the present worth of crash costs.

6.3 Comparison of Alternatives

The economic evaluation of each alternative results in the following information for each access control alternative: present worth of agency costs and present worth of user costs, which include operating costs and crash costs. The alternatives can then be compared to select the best one. An incremental approach can be used in which pairwise comparisons are made between alternatives. In this method, the projects are ranked in order of increasing agency costs. Pairwise comparisons of alternatives are then made, beginning with the alternative with the lowest agency costs. The incremental net present value between two projects can be calculated as

$$NPV_{ij} = \left(PWUC_i - PWUC_j\right) - \left(PWAC_j - PWAC_i\right) \tag{6.8}$$

where:

NPV_{ij} = incremental net present value between Alternatives i and j, where j is the alternative with the higher agency costs, and i is the current best alternative,

$PWUC_i$ = present worth of user costs for Alternative i,

$PWUC_j$ = present worth of user costs for Alternative j,

$PWAC_i$ = present worth of agency costs for Alternative i,

$PWAC_j$ = present worth of agency costs for Alternative j.

If the incremental net present value is greater than zero, the Alternative j is selected as the new current best alternative. After all comparisons are made, the final current best alternative is selected as the best access control alternative.

Table 6.1 Converted default parameters for TRANSYT-7F models (original values in Wallace et al. 1991b)

Parameter	Value
A_{11FC}	0.177079
A_{12FC}	-0.0023227
A_{13FC}	1.3683×10^{-5}
A_{21FC}	2.77244
A_{22FC}	0
A_{23FC}	0
A_{31FC}	0
A_{32FC}	0
A_{33FC}	8.9757×10^{-6}
A_{11OC}	78.815
A_{12OC}	0.10972
A_{13OC}	0.0029497
A_{14OC}	2.5000×10^{-5}
A_{21OC}	-0.59937
A_{22OC}	0.18821
A_{23OC}	0.012623
A_{24OC}	6.7176×10^{-5}
DC	145.1
FC	0.33
TC	1.3039
I	1
O	1.20

7. IMPLEMENTATION CONSIDERATIONS

This section describes the steps for use of the delay and safety models presented in this report and based on the framework presented in Chapter 3. A complete description of the procedure can be found in the *User's Guide*.

7.1 Procedure Framework

The analysis of each alternative is based on the operating costs and crash costs during the project lifetime as shown in Figure 7.1. The operating and crash costs may vary during each year of the project lifetime. Since the calculation of operating and crash costs for each year may become cumbersome, the project lifetime may be represented by a few typical periods as determined by abrupt changes in the traffic pattern. Each sub-period may be represented by typical years. For example, the first and last year of the sub-period could be selected as representative years, and linear interpolation could be used to find the operating and crash costs for other years. Another option would be to use the middle year of the sub-period as a representative year and assume that the annual costs are constant.

Once the turning volumes are predicted for each representative period and the impact area has been selected, each alternative is then evaluated. The analysis of each alternative includes estimating operating costs, estimating crash costs, and estimating agency costs. Finally, once each alternative is analyzed, the best access control alternative can be selected.

The calculation of user costs for a given representative year includes operating costs and crash costs.

(1) To calculate annual operating costs, a representative day may be selected. Each representative day may be represented by several typical hours such as morning hour, afternoon hour, and an hour representing the remainder of the day. The operating costs are then calculated for these typical hours based on the obtained turning volumes. The hourly operating costs can then be converted to daily operating costs and annual operating costs.

(2) Crash costs are estimated by first calculating the annual number of crashes by severity type using regression equations. The annual numbers of crashes are converted to crash costs by using default values for the costs of crashes by severity type.

7.2 Description of Modules

Module (6) Predict Operating Costs

The estimation of operating costs involves several steps as shown in Figures 7.2 to 7.6. Each representative day may be represented by several typical hours: morning hour, afternoon hour, and an hour representing the remainder of the day. For each typical hour, turning volumes are needed as input. These turning volumes can be obtained by running TRANPLAN. A daily traffic flow profile may be assumed to assist in the analysis.

The operating cost calculations are performed for each representative hour. These hourly operating costs are then converted into daily operating costs and annual operating costs. The following steps are needed for the calculation of operating costs:

- estimate total and average delays on arterial segments between intersections,
- correct the cruise speeds on arterial segments,
- perform calculation of operating cost (impact of access points not included),

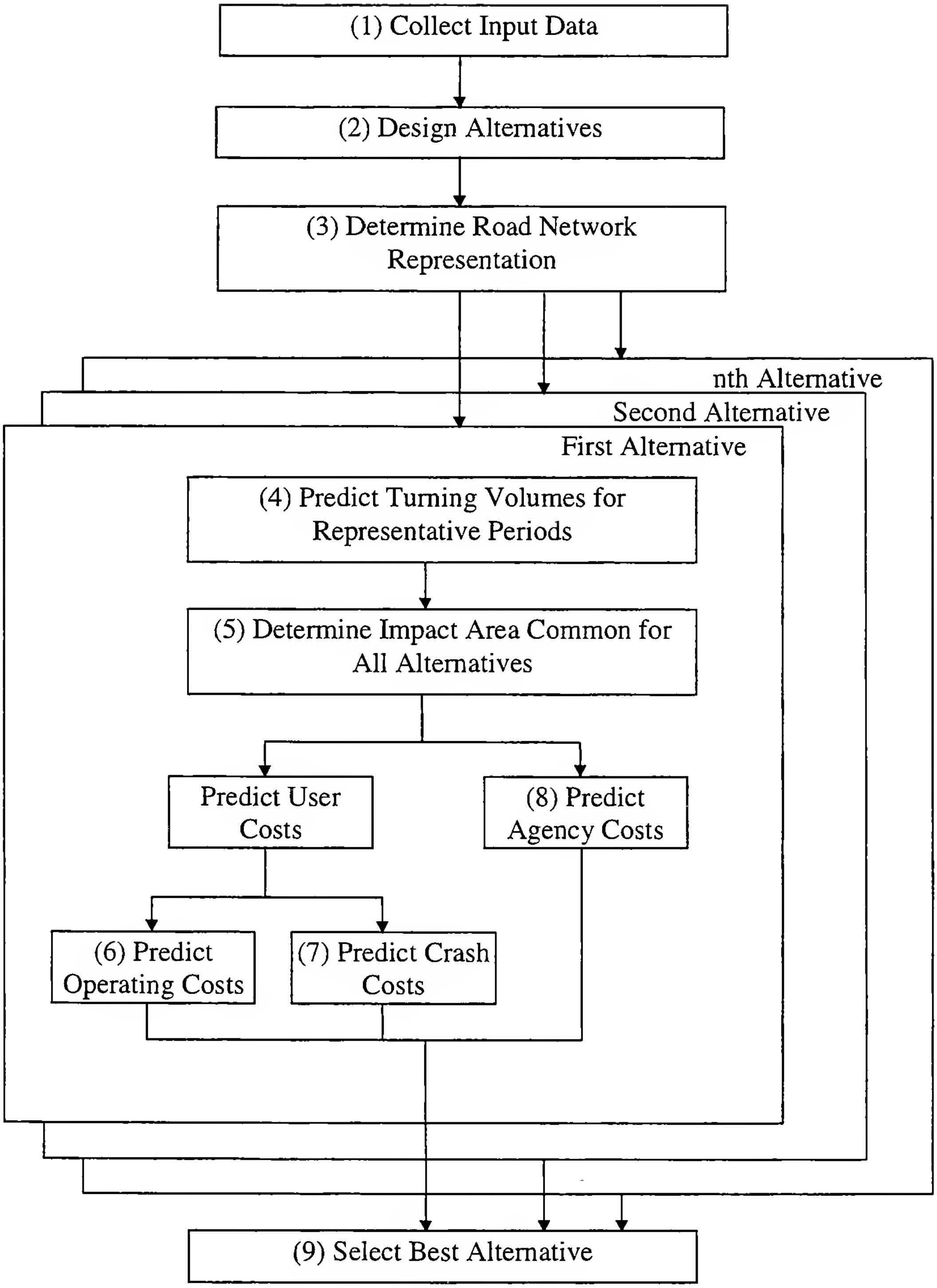

Figure 7.1 Procedure to evaluate access control alternatives

(6.1) Estimate total and average delays on arterial segments between intersections (Figure 7.3)

(6.2) Correct the cruise speeds on arterial segments using Equations 7.1 and 7.2

(6.3) Perform calculation of operating costs (impact of access points not included)

(6.4) Calculate extra operating costs caused by access points

(6.5) Aggregate hourly operating costs for the network

(6.6) Convert the obtained hourly costs into daily and annual operating costs

Figure 7.2 Procedure to estimate operating costs

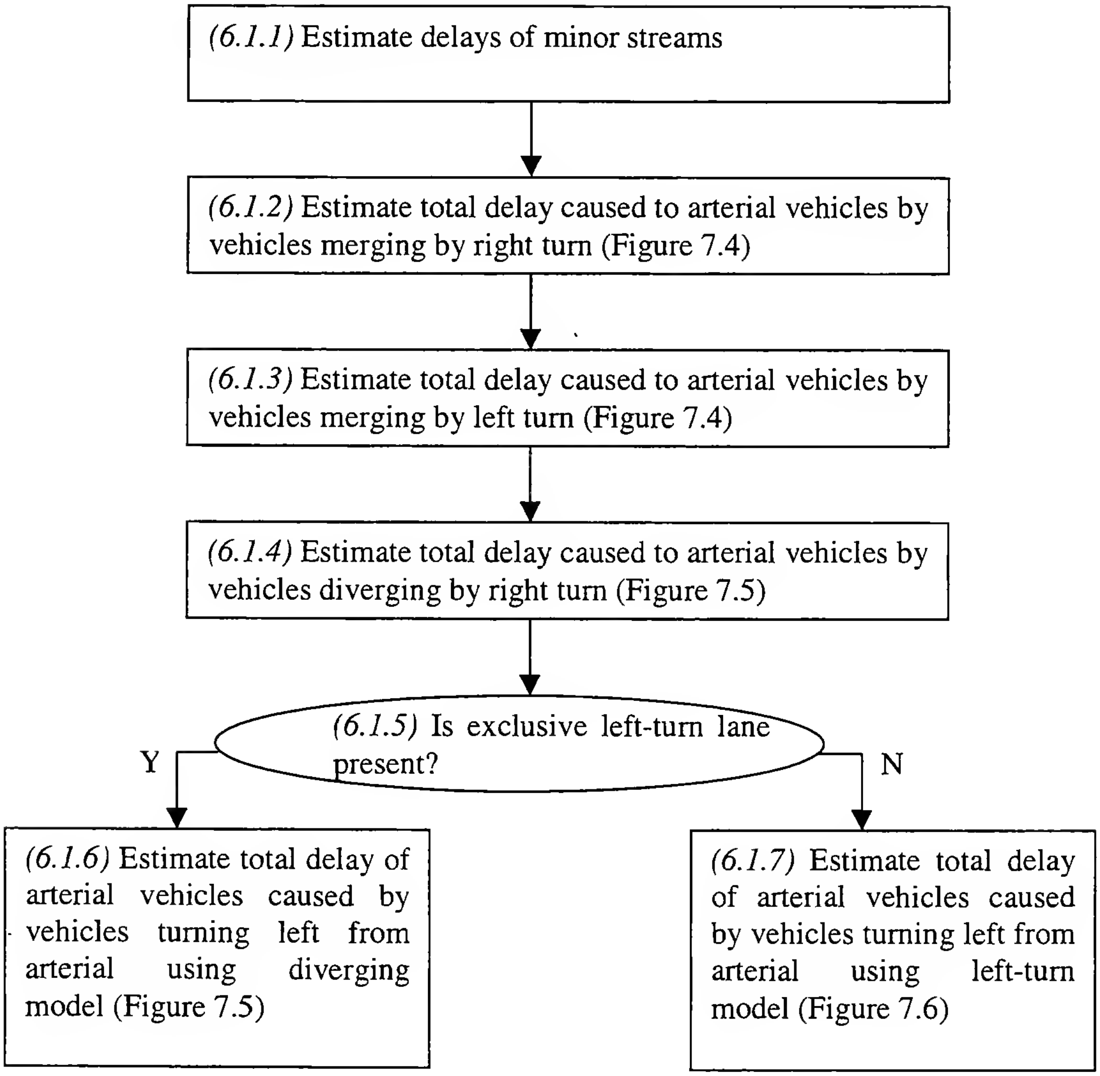

Figure 7.3 Procedure to estimate delays between intersections

(6.1.2.1) and (6.1.3.1) Assemble required input data to calculate delay caused by merging maneuver

↓

(6.1.2.2) and (6.1.3.2) Calculate t_m using Equations 4.3 and 4.4

↓

(6.1.2.3) and (6.1.3.3) Calculate g_i using Equation 4.6

↓

(6.1.2.4) and (6.1.3.4) Calculate $E(d_{1,nlc})$ using Equation 4.7

↓

(6.1.2.5) and (6.1.3.5) Calculate P_s using Equation 4.8

↓

(6.1.2.6) and (6.1.3.6) Calculate $E(d_1)$ using Equation 4.9

↓

(6.1.2.7) and (6.1.3.7) Calculate average delay of other vehicles using Equations 4.12 to 4.17

↓

(6.1.2.8) and (6.1.3.8) Add $E(d_1)$ and average delay of other vehicles to obtain average delay caused by one merging maneuver

↓

(6.1.2.9) and (6.1.3.9) Multiply average delay caused by one merging maneuver by merging volume to obtain total delay caused by merging

Figure 7.4 Procedure to estimate delay caused by merging maneuver

(6.1.4.1) and (6.1.6.1) Assemble required input data to calculate delay caused by diverging maneuver

(6.1.4.2) and (6.1.6.2) Calculate t_d using Equations 4.18 and 4.19

(6.1.4.3) and (6.1.6.3) Calculate g_i using Equation 4.34 and g_s using Equation 4.28

(6.1.4.4) and (6.1.6.4) Calculate $E(g/g_s<g<g_i)$ using Equation 4.40

(6.1.4.5) and (6.1.6.5) Calculate $E(d_{1,a})$ using Equation 4.27

(6.1.4.6) and (6.1.6.6) Calculate $P_{i,a}$ using Equation 4.29 and $P_{i,b}$ using Equation 4.39

(6.1.4.7) and (6.1.6.7) Estimate $E(d_{1,b}/d_{1,b}>0)$ by substituting $E(g/g_s<g<g_i)$ into Equations 4.33 and 4.38

(6.1.4.8) and (6.1.6.8) Calculate $E(d_{1,nlc})$ using Equation 4.41

(6.1.4.9) and (6.1.6.9) Calculate P_s using Equation 4.8

(6.1.4.10) and (6.1.6.10) Calculate $E(d_1)$ using Equation 4.42

(6.1.4.11) and (6.1.6.11) Calculate average delay of other vehicles using Equations 4.12 to 4.17

(6.1.4.12) and (6.1.6.12) Add $E(d_1)$ and average delay of other vehicles to obtain average delay caused by one diverging maneuver

(6.1.4.13) and (6.1.6.13) Multiply total average delay caused by one diverging maneuver by diverging volume to obtain total delay caused by diverging

Figure 7.5 Procedure to estimate delay caused by diverging maneuver

(6.1.7.1) Assemble required input data to calculate delay caused by left-turn maneuver

↓

(6.1.7.2) Calculate t_l from Equation 4.43

↓

(6.1.7.3) Calculate P_{s2} from Equation 4.45

↓

(6.1.7.4) Calculate h_r from Equation 4.48

↓

(6.1.7.5) Calculate r from Equations 4.46 and 4.47

↓

(6.1.7.6) Calculate P_j from *Equation* 4.44

↓

(6.1.7.7) Calculate d_t from Equation 4.49

↓

(6.1.7.8) Multiply d_t by the left-turning volume to obtain the total delay caused by left-turning vehicles

Figure 7.6 Procedure to estimate delays caused by left turn from arterial when left-turn lane is not present

- calculate extra operating costs caused by access points, and
- aggregate hourly operating costs for the network.

The following paragraphs discuss these steps in more detail.

Module (6.1) Estimate total and average delays on arterial segments between intersections

The delay models presented in Chapter 4 can be used to estimate delays between intersections. These delays include the delays of minor streams at access points and the delays of arterial streams caused by minor streams at access points. These calculations can be performed using a Microsoft Excel spreadsheet. The obtained turning volumes are used as input to the delay and operating costs calculations.

Module (6.1.1) Estimate delays of minor streams

Delays of the minor streams can be calculated for the following minor stream maneuvers on each segment:

- right turn onto the arterial,
- left turn from the arterial, and
- left turn onto the arterial.

The capacities of these movements can be calculated using the *Highway Capacity Manual* (1994). Once the capacity of each maneuver is calculated, the average delay for each maneuver is calculated, assuming no queuing at access points. The total delay for each minor stream maneuver is then calculated by multiplying the average delay of the maneuver by the number of maneuvers of a given type. The total delay experienced by all minor streams for a given segment is then calculated by adding the total delay experienced by each minor stream.

It is assumed in the calculations that the minor streams do not experience queuing. If there is an access point on the segment where minor streams experience queuing, the access point should be represented as an intersection in the network.

Module (6.1.2) Estimate total delay caused to arterial vehicles by vehicles merging by right turn

The merging model is used to estimate the total delay of through vehicles caused by vehicles turning right onto the arterial. Figure 7.4 summarizes the sequence of calculations. The following input data for each link are needed for the merging model:

- right-turn merging speed,
- right-turn merging distance,
- right-turn merging volume,
- arterial speed, and
- arterial traffic distribution by lane.

The merging distances can be estimated for given merging and arterial speeds using Figure II-16 in AASHTO (1994). The model parameters and assumed default values include

- saturation flow rate (0.5 veh/sec),
- lag time (4 sec),
- critical gap for merging (5.5 sec from *Highway Capacity Manual*),
- critical space gap for lane changing (4 sec × arterial speed),
- likelihood that a driver wants to change lanes (0.5), and
- proportion of traffic in each lane (equal distribution).

These values can be modified by the user as needed. The output of this step includes the total delay caused by right-turn merging maneuvers on each segment.

Module (6.1.3) Estimate total delay caused to arterial vehicles by vehicles merging by left turn

The procedure for estimating the total delay caused to through vehicles caused by vehicles turning left onto the arterial involves calculations similar to those used in *Module (6.1.2)*. The input data requirements include

- left-turn merging speed,
- left-turn merging distance,
- left-turn merging volume,

- arterial speed, and
- arterial traffic distribution by lane.

The output of this step includes the total delay caused by left-turn merging maneuvers on each segment.

Module (6.1.4) Estimate total delay caused to arterial vehicles by vehicles diverging by right turn

The diverging model is used to estimate the total delay of through vehicles caused by vehicles turning right from the arterial. Figure 7.5 summarizes the sequence of calculations. The following input data for each link are needed for the right-turn diverging model:

- right-turn diverging speed,
- right-turn diverging distance,
- right-turn diverging volume,
- arterial speed, and
- arterial traffic distribution by lane.

The diverging distances can be estimated for given diverging and arterial speeds using Figure II-17 in AASHTO (1994). The model parameters and assumed values include

- saturation flow rate (0.5 veh/sec),
- critical space gap for lane changing (4 sec × arterial speed),
- reaction time for through vehicle (1.5 sec),
- acceleration rate of through vehicle (1.5 m/sec^2),
- maximum allowable deceleration rate (-4.9 m/sec^2),
- likelihood that a driver wants to change lanes (0.5), and
- proportion of traffic in each lane (equal distribution).

These values can be modified by the user as needed. The output of this step includes the total delay caused by right-turn diverging maneuvers on each segment.

Module (6.1.5) Check for presence of exclusive left-turn lane

The presence of an exclusive left-turn lane determines the model used to estimate the delay caused by vehicles turning left from the arterial. If a left-turn lane is present, the diverging model should be used to estimate the delay caused by left-turning vehicles. If a left-turn lane is not present, the left-turn model should be used to estimate the delay caused by left-turning vehicles.

Module (6.1.6) Estimate total delay of arterial vehicles caused by vehicles turning left from arterial using diverging model (when left-turn lane is present)

This procedure is similar to the procedure in *Module (6.1.4)* with different input data values. The following input data for each link are needed for the left-turn diverging model:

- left-turn diverging speed,
- left-turn diverging distance,
- left-turn diverging volume,
- arterial speed, and
- arterial traffic distribution by lane.

The diverging distances can again be estimated using Figure II-17 in AASHTO (1994). The output of this step includes the total delay caused by left-turn diverging maneuvers on each segment.

Module (6.1.7) Estimate total delay of arterial vehicles caused by vehicles turning left from arterial using left-turn model (when left-turn lane is not present)

When a left-turn lane is not present, the left-turning vehicle may block through vehicles. In this case, the left-turn model is used to estimate the delay caused to through vehicles by vehicles turning left from the arterial. Figure 7.6 shows the steps in performing the calculations. The following input data are needed:

- arterial speed,
- blockage time of left-turning vehicle, and
- sight distance when the left-turning vehicle is first observed by the through vehicle.

The model parameters and default values include

- speed reduction when through vehicle approaches left-turning vehicle (15 km/hr),
- critical space gap for lane change (4 sec × arterial speed),
- likelihood that a driver wants to change lanes (1.0), and
- saturation flow rate (0.5 veh/sec).

These parameter values can be modified as needed. The output includes the total delay caused to through vehicles by left-turning vehicles.

Module (6.2) Correct the cruise speeds on arterial segments

Once the delays caused by minor streams are estimated, the initial cruise speeds provided by the user are then adjusted to account for the reduction in travel time associated with the delays caused by minor streams. The initial cruise speeds used to calculate the delays at access points are adjusted for each link before estimating delays at intersections. To revise the cruise speed, the average delay for each through vehicle needs to be estimated as

$$d_{t,i} = \frac{D_{t,i}}{Q_{t,i}} \tag{7.1}$$

where:

$d_{t,i}$ = average delay experienced by one arterial vehicle on link i (sec/veh),

$D_{t,i}$ = total delay caused by minor streams on link i (sec/hr),

$Q_{t,i}$ = total volume on link i (vph).

The revised cruise speed for each link can then be estimated:

$$V_i' = V_i^0 \cdot \left(\frac{1}{1 + \frac{d_{t,i} \cdot V_i^0}{3600 \cdot L_i}} \right) \tag{7.2}$$

where:

V_i^0 = initial cruise speed on link i (km/hr),

V_i' = revised cruise speed on link i (km/hr),

L_i = length of link i (km).

Module (6.3) Perform calculation of operating cost (impact of access points not included)

The TRANSYT-7F software can be used to optimize signal timings and to calculate delays, stops, and operating costs for the network not including access points. The value of the operating costs for the network is one of the measures of effectiveness provided in the TRANSYT-7F output. Thus, TRANSYT-7F software can be used to directly estimate operating costs for the network not including the effects of access points. The operating costs calculated by TRANSYT-7F are given in 1987 dollars. The user should multiply the operating costs as determined by TRANSYT-7F by the appropriate inflation factor to convert the operating costs to the chosen base year. To estimate delays and operating costs, the turning volumes for intersections need to be provided as input to TRANSYT-7F. The corrected cruise speed for each link is also needed as input to TRANSYT-7F.

One existing software package that could be used to manage the data for intersections is SYNCHRO3. SYNCHRO3 provides a graphical interface that can be used to enter network and traffic data. SYNCHRO3 can also be used to optimize phases at signalized intersections. SYNCHRO3 can convert the input data to TRANSYT-7F

formats, and TRANSYT-7F can be run directly from SYNCHRO3. The TRANSYT-7F data editor may also be used to enter the network and traffic data.

Other software than TRANSYT-7F can be used to estimate the delays and stops at intersections. The delays, stops, and travel times should be converted to operating costs using the TRANSYT-7F models (Wallace et al., 1991b) as described in Chapter 6. The TRANSYT-7F operating cost model includes fuel consumption and the cost of travel time, delay, stops, and total travel. The user should select the appropriate inflation factor I to convert the operating costs from 1987 dollars to the chosen base year.

Module (6.4) Calculate extra operating costs caused by access points

The extra operating costs caused by access points are then calculated based on the equations given by Wallace et al. (1991b). The total delay caused to arterial vehicles by all minor streams and total delay of minor streams are needed as input. The equations to calculate the additional operating costs caused by access points are given by

$$F_a = K_{2,FC} \cdot D_a \tag{7.3}$$

$$F_m = K_{2,FC} \cdot D_m \tag{7.4}$$

$$C_{ma} = \left(F_a \cdot FC + \frac{DC \cdot D_a}{1000} \right) \cdot I + \left(F_m \cdot FC + \frac{DC \cdot D_m}{1000} + O \cdot TC \cdot D_m \right) \cdot I \tag{7.5}$$

where:

F_a = additional fuel consumption for arterial vehicles caused by access points (lit),

F_m = additional fuel consumption of minor stream vehicles (lit),

D_a = total delay to arterial vehicles caused by all minor stream maneuvers (veh-hr),

D_m = total delay of minor streams at access points (veh-hr),

C_{ma} = additional operating cost due to access points.

The appropriate inflation factor I should be selected for use in Equation 7.5.

Module (6.5) Aggregate hourly operating costs for the network

The extra operating costs caused by access points are added to the operating cost with the impact of access points not included to obtain the total hourly operating cost. The results for a given hour are aggregated for the network.

Module (6.6) Convert the obtained hourly costs into daily and annual operating costs

The hourly operating costs need to be combined to obtain daily operating costs. The delay analysis is performed for three typical hours: a morning hour, an afternoon hour, and an hour representing the rest of the day. A daily traffic flow profile may be assumed to assist in the analysis. The hourly operating costs for the three typical hours can be converted to daily operating costs as follows:

$$C_{day} = N_{am} \cdot C_{am} + N_{pm} \cdot C_{pm} + (24 - N_{am} - N_{pm}) \cdot C_{op} \qquad (7.6)$$

where:

C_{day} = daily operating cost,

C_{am} = operating cost for morning hour,

C_{pm} = operating cost for afternoon hour,

C_{op} = operating cost for hour representing remainder of day,

N_{am} = number of morning hours,

N_{pm} = number of afternoon hours.

Once daily operating costs have been estimated, they can be converted to annual operating costs by multiplying the daily operating costs by the number of days per year. The final result of this step is the annual operating costs of a given access control alternative for a typical year.

Module (7) Predict Crash Costs

In addition to operating costs, the crash costs also need to be estimated. Crash rates for each alternative are estimated for intersections and for segments between

intersections. The crash rates are then aggregated for the entire network and converted to crash costs by using the unit costs of crashes by severity type. Figures 7.7 to 7.10 summarize the steps for predicting crash rates.

Module (7.1) Estimate annual number of crashes for multi-lane arterial segments

Crash rates for multi-lane arterial segments are calculated using the regression models in Equations 5.2 to 5.4. For each segment, the required input includes

- segment length,
- number of years,
- AADT,
- access density,
- proportion of access points that are signalized,
- presence of outside shoulder,
- presence of two-way left-turn lane, and
- presence of a median with no openings between signals.

The segments should be homogeneous with respect to AADT, cross section, and level of access control. The AADT data can be obtained by running transportation planning software. The number of crashes by severity type is then calculated. The annual number of crashes may be found by setting the value of *YRS* to one. The number of property-damage-only crashes can be calculated from Equation 5.3, and the number of fatal/injury crashes can be calculated from Equation 5.4. Crash rates for two-lane arterial segments should be calculated using the procedure in *Module (7.2)*.

Module (7.2) Estimate annual number of crashes for other segments in impact area

The annual number of crashes for other segments in the impact area is estimated by using the basic calibrated models developed by Eranky et al. (1997). The required input for these models includes segment length and AADT. The number of crashes by severity type on urban multi-lane segments can be calculated using Equations 5.6 and 5.7.

The annual number of crashes for urban two-lane segments by severity type can be calculated using Equations 5.9 and 5.10.

Module (7.3) Estimate annual number of crashes for signalized intersections in impact area

Crash rates for signalized intersections located at the endpoints of the arterial segments or outside the arterial but in the impact area can also be estimated using the models developed by Jonathan Weiss. The annual number of crashes by severity type at signalized intersections can be calculated using Equations 5.11 and 5.12. The input data requirements for these models include

- average volume on N-S approaches,
- average volume on E-W approaches,
- number of intersection approaches,
- number of left-turning movements forbidden on all approaches, and
- number of approaches on which traffic is divided by a median.

Module (7.4) Aggregate crashes by severity level

The crash rates for the segments and intersections in the impact area are then aggregated by severity type. The final result is the annual number of property-damage-only crashes and the annual number of fatal/injury crashes for each typical year for a given access control alternative.

Module (7.5) Calculate annual total cost of crashes

The annual total cost of crashes is calculated by using the predicted number of crashes and the unit costs of crashes by severity level. The annual total cost of property-damage-only crashes is found by multiplying the annual number of property-damage-only crashes by the unit cost of a property-damage-only crash. The annual total cost of fatal/injury crashes is found by multiplying the annual number of fatal/injury crashes by the unit cost of a fatal/injury crash. The annual total cost of crashes for a given year and a given access control alternative is then found by adding the annual total cost of property-

(7.1) Estimate annual number of crashes for multi-lane arterial segments (Figure 7.8)

↓

(7.2) Estimate annual number of crashes for other segments in impact area (Figure 7.9)

↓

(7.3) Estimate annual number of crashes for signalized intersections in impact area (Figure 7.10)

↓

(7.4) Aggregate crashes by severity level

↓

(7.5) Calculate annual total cost of crashes

Figure 7.7 Procedure to estimate crash costs

(7.1.1) Assemble input data for each multi-lane arterial segment: *LEN*, *YRS*, *AADT*, *ACCESS*, *SHLDR*, *PS*, *TWLTL*, *NOMEDO*

↓

(7.1.2) Estimate number of pdo crashes for each multi-lane arterial segment using Equation 5.3

↓

(7.1.3) Estimate number of fatal/injury crashes for each multi-lane arterial segment using Equation 5.4

Figure 7.8 Procedure to estimate crashes for multi-lane arterial segments

(7.2.1) Assemble input data for each segment: *LEN*, *AADT*

(7.2.2) Estimate number of pdo crashes on urban multi-lane segments using Equation 5.6

(7.2.3) Estimate number of fatal/injury crashes on urban multi-lane segments using Equation 5.7

(7.2.4) Estimate number of pdo crashes on urban two-lane segments using Equation 5.9

(7.2.5) Estimate number of fatal/injury crashes on urban two-lane segments using Equation 5.10

Figure 7.9 Procedure to estimate crashes for two-lane segments in impact area and multi-lane segments outside arterial

(7.3.1) Assemble input data for each signalized intersection: *APPVOL1*, *APPVOL2*, *NUMAPP*, *LFTPROB*, *NUMDIV*

(7.3.2) Estimate number of pdo crashes for each signalized intersection using Equation 5.11

(7.3.3) Estimate number of fatal/injury crashes for each signalized intersection using Equation 5.12

Figure 7.10 Procedure to estimate crashes for signalized intersections in impact area

damage-only crashes and the annual total cost of fatal/injury crashes. The final result of this step is annual total cost of crashes for each representative year under a given access control alternative.

Module (8) Predict Agency Costs

The agency costs also need to be estimated. The agency costs for a given alternative include the construction and maintenance costs of the alternative. These costs can be converted to present worth.

Module (9) Select Best Alternative

Once each access control alternative has been evaluated for user costs and agency costs, the alternatives can be compared to select the best one. An incremental approach based on pairwise comparisons of alternatives can be used. After comparing the alternatives, the best access control alternative can be selected.

8. CONCLUSIONS

The objective of this research was to develop a comprehensive procedure to evaluate access control alternatives. The procedure is based on a quantitative assessment of the user costs and agency costs of each alternative. The procedure involves collecting input data, designing the alternatives, determining the road network, evaluating each alternative, and selecting the best alternative. Evaluation of each alternative includes predicting operating costs, crash costs, and agency costs. After the alternatives are compared and evaluated, the best alternative can be selected.

8.1 Delay Models

Prediction of operating costs involves estimating traffic delays and stops for intersections and segments between intersections. Many models and software tools exist to estimate delays at signalized intersections, such as HCS, TRANSYT-7F, and PASSER II. Many models also exist to estimate delays for minor streams at unsignalized intersections.

Models are needed to estimate delays of arterial vehicles caused by minor streams at unsignalized intersections. Such models were developed as a part of this study. These analytical models can be used to predict delays of arterial vehicles caused by merging, diverging, and left-turn maneuvers. The input data requirements for the models include arterial speeds, turning speeds, arterial volumes, and turning volumes. For the merging and diverging maneuvers, the sensitivity analyses showed that the average delay increases as the arterial volume and arterial speed increase. The average delay decreases

as the turning speed increases. If lane changing is incorporated, the average delay decreases. The effect of lane changing becomes less significant at higher arterial volumes because there are fewer opportunities to change lanes. For the left-turn maneuver with a shared lane, the results of the sensitivity analyses showed that the average delay increases as the arterial volume and blockage time increase. Lane changing can also reduce the delay caused by the left-turn maneuver. The developed models can be used to estimate the delays between intersections.

To estimate the operating costs during the project lifetime, a few representative years may be selected. Each representative year may be represented by a typical day. Each typical day may be represented by a few hours: morning hour, afternoon hour, and an hour representing the remainder of the day. Once the hourly delays and stops have been estimated for a given access control alternative, the operating costs need to be calculated. The TRANSYT-7F models can be used to convert the hourly delays and stops to operating costs. The hourly operating costs can then be converted to daily operating costs by assuming a daily flow profile. The daily operating costs can then be converted to annual operating costs.

8.2 Safety Models

In addition to operating costs, crash costs also need to be estimated. Models were developed in this study to predict crash rates on multi-lane arterial segments based on geometric and access control characteristics. The models for total number of crashes, property-damage-only crashes, and fatal/injury crashes all have the same structure. The exposure to risk variables include segment length, AADT, and number of years. The significant factors include access density, proportion of signalized access points, presence of an outside shoulder, presence of a two-way left-turn lane, and presence of a median with no openings between signalized intersections. The number of crashes increases as the access density and proportion of signalized access points increase. An outside shoulder, two-way left-turn lane, or median without openings between signals leads to a

reduction in the number of crashes. These models can be used to estimate crash rates on multi-lane arterial segments between intersections. Other models can be used to predict crash rates for other segments in the impact area and for signalized intersections. The annual number of crashes by severity type can be converted to annual crash costs by using unit values for the costs of property-damage-only and fatal/injury crashes.

Once the crash costs and operating costs for each alternative have been calculated, they can be converted to present worth. The agency costs for each alternative can also be estimated and converted to present worth. The alternatives can then be compared to select the best one.

LIST OF REFERENCES

AASHTO. *A Policy on Geometric Design of Highways and Streets.* Washington, D.C., 1994.

Agent, K.R. Warrants for Left-Turn Lanes. *Transportation Quarterly.* Volume 37. January 1983.

Alexander, M. H. *Development of an Economic Warrant for the Construction of Right-Turn Deceleration Lanes.* Final Report. Joint Highway Research Project C-36-17HH. Purdue University. West Lafayette, Indiana, 1970.

Azzeh, J.A., Thorson, B.A., Valenta, J.J., Glennon, J.C., and Wilton, C.L. *Evaluation of Techniques for the Control of Direct Access to Arterial Highways.* Report FHWA-RD-76-85. Federal Highway Administration. Washington, D.C., 1975.

Ballard, J.L., and McCoy, P.T. Operational Effects of Two-Way Left-Turn Lanes on Two-Way, Four-Lane Streets. *Transportation Research Record 923.* Transportation Research Board. Washington, D.C., 1983.

Ballard, J.L., and McCoy, P.T. Computer Simulation Study of the Operational Effects of Two-Way Left-Turn Lanes on Urban Four-Lane Roadways. *Transportation Research Record 1195.* Transportation Research Board. Washington, D.C., 1988.

Bonneson, J.A., and Fitts, J.W. Delay to Major-Street Through Vehicles at Two-Way Stop Controlled Intersections. Proceedings of the Third International Symposium on Intersections Without Traffic Signals. Portland, Oregon. July 1997.

Eranky, S., Tarko, A.P., and Sinha, K.C. *Crash Reduction Factors for Improvement Activities on Road Sections in Indiana.* Draft Final Report. Purdue University. West Lafayette, Indiana, August 1997.

Garber, N.J., and White, T.E. Guidelines for Commercial Driveway Spacing on Urban and Suburban Arterial Roads. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Gattis, J.L. Comparison of Delay and Accidents on Three Roadway Access Designs in a Small City. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Highway Capacity Manual. TRB Special Report 209. National Research Council. Washington, D.C., 1994.

Harmelink, M.D. Volume Warrants for Left-Turn Storage Lanes at Unsignalized Grade Intersections. *Highway Research Record 211.* Highway Research Board. Washington, D.C., 1967.

Hawley, P.E., and Stover, V.G. Guidelines for Left-Turn Bays at Unsignalized Access Locations. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Hummer, J.E., and Boone, J.L. Travel Efficiency of Unconventional Suburban Arterial Intersection Designs. *Transportation Research Record 1500.* Transportation Research Board. Washington, D.C., 1995.

Kyte, M., Tian, Z., Mir, Z., Hameedmansoor, Z., Kittelson, W., Vandehey, M., Robinson, B., Brilon, W., Bondzio, L., Wu, N., and Troutbeck, R. NCHRP Project 3-46: *Capacity and Level of Service at Unsignalized Intersections-Final Report.* Vol. 1. April 1996.

Lall, B.K., Huntington, D., and Eghtedari, A. Access Management and Traffic Safety. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Li, J., Abdelwahab, W., and Brown, G. Joint effects of access and geometry on two-lane rural highway safety in British Columbia. *Canadian Journal of Civil Engineering.* December 1994.

McCoy, P.T., Ballard, J.L., Eitel, D.S., and Witt, W.E. Two-Way Left-Turn Lane Guidelines for Urban Four-Lane Roadways. *Transportation Research Record 1195.* Transportation Research Board. Washington, D.C., 1988.

McCoy, P.T., Ballard, J.L., and Wiyaya, Y.H. Operational Effects of Two-Way Left-Turn Lanes on Two-Way, Two-Lane Streets. *Transportation Research Record 869.* Transportation Research Board. Washington, D.C., 1982.

McCoy, P.T., Bonneson, J.A., Ataullah, S., and Eitel, D.S. Guidelines for Right-Turn Lanes on Urban Roadways. *Transportation Research Record 1445*. Transportation Research Board. Washington, D.C., 1994.

McCoy, P.T., and Malone, M.S. Safety Effects of Left-Turn Lanes on Urban Four-Lane Roadways. *Transportation Research Record 1239.* Transportation Research Board. Washington, D.C., 1989.

McGuirk, W. W. *Evaluation of Factors Influencing Driveway Accidents.* Master's Thesis. Purdue University. May 1973.

McShane, W.R., Choi, D.S., Eichin, K., and Sokolow, G. Insights Into Access Management Details Using TRAF-NETSIM. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Olson, P.L., Cleveland, D.E., Fancher, P.S., Kostynuik, L.P., and Schneider, L.W. *Parameters Affecting Stopping Sight Distance.* NCHRP Report 270. Highway Research Board, Washington, D.C., 1984. (Developed by New York DOT. from report).

Poe, C.M., Tarris, J.P., and Mason, J.M. Influence of Access and Land Use on Vehicle Operating Speeds Along Low-Speed Urban Streets. Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Ring, S.L., and Carstens, R.L. Guidelines for the Inclusion of Left-Turn Lanes at Rural Highway Intersections. *Highway Research Record 371.* Highway Research Board. Washington, D.C., 1970.

Squires, C.A., and Parsonson, P.S. Accident Comparison of Raised Median and Two-Way Left-Turn Lane Median Treatments. *Transportation Research Record 1239.* Transportation Research Board. Washington, D.C., 1989.

Stover, V.G., Adkins, W.G., and Goodknight, J.C. *Guidelines for Medial and Marginal Access Control on Major Roadways.* NCHRP Report 93. Highway Research Board. Washington, D.C., 1970.

Uckotter, D.A. *Analysis of Motor Vehicle Accidents at Commercial Driveway Locations.* Master's Thesis. Purdue University. May 1974.

Vargas, F., and Reddy, G.V. Does Access Management Improve Traffic Flow? Can NETSIM Be Used to Evaluate? Conference Proceedings. 2nd National Access Management Conference. Vail, Colorado. August 1996.

Wallace, C.E., Chang, E., Messer, C.J., and Courage, K.G. *PASSER II-90 Users Guide.* Methodology for Optimizing Signal Timing: Volume 3. Federal Highway Administration. Washington, D.C., 1991a.

Wallace, C.E., Courage, K.G., and Hadi, M.A. *TRANSYT-7F Users Guide.* Methodology for Optimizing Signal Timing: Volume 4. Federal Highway Administration. Washington, D.C., 1991b.

Weiss, J. *Identification of Hazardous Intersections Based on Potential Crash Reduction.* Master's Thesis. Purdue University. December 1996.

APPENDIX A: List of Access Control Techniques

APPENDIX A: List of Access Control Techniques

Based on the report by Azzeh et al. (1975) and on revisions by INDOT.

Technique	Effects	Affect traffic pattern
install median barrier	Reduce conflict points	X
install raised median divider with left-turn deceleration lanes	Reduce conflict points	
install one-way operations on highway	Reduce conflict points	X
install traffic signal	Reduce conflict points	
channelize median openings to prevent left-turn ingress or egress maneuvers	Reduce conflict points	X
offset opposing driveways or public road approaches	Reduce conflict points	
locate driveway opposite 3-leg intersection and install signal	Reduce conflict points	
install two one-way driveways in lieu of one two-way driveway	Reduce conflict points	
install two two-way driveways with limited turns in lieu of one standard two-way driveway	Reduce conflict points	
install two one-way driveways in lieu of two two-way driveways	Reduce conflict points	
install two two-way driveways with limited turns in lieu of two standard two-way driveways	Reduce conflict points	
install driveway channelizing island to prevent left turns	Reduce conflict points	X
install grade separation structure	Reduce conflict points	X
install interchange	Reduce conflicts	X
install blvd left turns	Reduce conflicts	X
eliminate access point	Reduce conflicts	X
regulate minimum spacing of driveways	Reduce frequency and severity of conflict points	X
regulate minimum corner clearance	Restrict access Increase perception time Increase deceleration distance	X

Technique	Effects	Affect traffic pattern
optimize driveway spacing	Restrict access Increase perception time Increase deceleration distance	X
regulate maximum number of driveways per frontage	Reduce frequency and severity of conflict points	X
consolidate access	Reduce frequency and severity of conflict points	X
buy abutting properties	Reduce frequency and severity of conflict points	X
deny access to small frontages	limit number of access points increase perception time and distance	X
consolidate existing access	lessen deceleration requirements	X
designate number of driveways	lessen deceleration requirements	X
require access on collector	reduce frequency and severity of conflict points	X
encourage connections between adjacent properties	reduce frequency and severity of conflict points	
improve driveway sight distance	increase perception, reaction, and braking distances	
regulate minimum sight distance	increase perception, reaction, and braking distances reduce speed differential	
optimize sight distance in permit authorization stage	increase perception, reaction, and braking distances reduce speed differential	
increase driveway effective approach width	increase driveway turning speeds	
improve driveway vertical geometrics	increase driveway turning speeds	
install right-turn acceleration lane	increase speed for merging vehicles	
install channelizing islands to move ingress merge point laterally away from highway	increase driveway turning speeds	
install turning roadway (increase radii)	increase driveway turning speeds	
install two-way left-turn lane	remove left-turning vehicles from through lanes	

Technique	Effects	Affect traffic pattern
install isolated median and deceleration lane for left-turning vehicles	remove left-turning vehicles from through lanes	
install left-turn deceleration lane in lieu of right-angle crossover	remove left-turning vehicles from through lanes	
install medial storage for left-turn egress vehicles	remove left-turning vehicles from through lanes	
increase storage capacity of existing left-turn lane	remove left-turning vehicles from through lanes	
install continuous right-turn lane	remove right-turning vehicles at higher speeds	
install right-turn deceleration lane	remove turning vehicles from through lanes	
install supplementary one-way right-turn driveways to divided highway	removing turning vehicles and queues from through lanes	
provide U-turn maneuver at designated areas	remove turning vehicles from through lanes	X
construct local service or frontage road	remove turning vehicles from through lanes	X
install supplementary access on collector	remove turning vehicles or queues from the through lanes	X
install additional driveway when total driveway demand exceeds capacity	Reduce length of queues waiting to enter driveway	
install additional exit lane on driveway	Decrease driveway delay	

APPENDIX B: Description of Crash Extraction Software

APPENDIX B: Description of Crash Extraction Software

The crash extraction software uses two data files as input: the Crash Database File and the County File. The software produces two output files that contain the number of crashes on each of the segments and the summary statistics regarding the number of crashes missed. Based on the developed algorithm to extract crashes, the computer program was written in C++ by Nakarin Satthamnuwong, a graduate student at Purdue.

Format of Crash Database File

The formatted Crash Database File contains six fields extracted from the Crash Database records. The formatted file is a space-delimited text file. Each line corresponds to one crash in Indiana for a given year. Each line contains six fields in this order: *county*, *severity*, *road*, *roadref*, *dirref*, and *distref*. The following is a brief description of each of these fields:

county	The number of the county in which the crash occurred.
severity	The crash severity.
road	The pseudo number of the road on which the crash occurred. A missing main street pseudo number is indicated by a value of 0. A missing main street pseudo number can occur when the street name in the accident report does not have a corresponding pseudo number.
roadref	The pseudo number of the cross street from which the crash direction and distance are measured. A missing cross street pseudo number is indicated by a value of 0. A missing cross street pseudo number can occur when the cross street name in the accident report does not have a corresponding pseudo number. The cross street may have been misspelled or may be a commercial driveway that does not have a corresponding pseudo number.
dirref	The direction from the reference intersection where the crash occurred. Missing direction is indicated with a period. Missing direction with a *distref* value of 0 indicates that the crash occurred at an intersection.

Missing direction with any other value of *distref* indicates that the crash cannot be located.

distref The distance from the reference intersection (feet) where the crash occurred. Unknown distance is indicated with a period. An unknown distance indicates that the crash cannot be located. A distance of 0 indicates that the crash occurred at an intersection.

Format of County File

The files obtained from running the RIDB matching software developed by Weiss (1996) need to be formatted before they can used with the crash software. The formatting can be done using a Microsoft Excel macro developed by Eranky et al. (1997). The resulting files are hereafter referred to as the County Files. The fields from the formatted County File that are used by the program are listed in Table B.1. The coding for the directions in column 17 must be done manually. The numbers used for coding the directions should be the same as those used in the Crash Database File.

Table B.1 Fields in County File used by program

Column Number	Description
1	DRK number of subsegment
6	Length of subsegment (hundredths of mile)
8, 9, 10, 11	Main street pseudo numbers of subsegment
12, 13, 14, 15, 16	Cross street pseudo numbers
17	Geographic direction in which subsegments are listed when moving down in the County File

Running the Program

To run the program, the County File and Crash Database File should be in the same directory as the program. The program can be executed at the DOS prompt by typing

crash1 Co_No Int_File CDB_File

where:

Co_No = Number of county for which crashes are being extracted.

Int_File = Name of the County File. The file must have a *.txt* extension, but the extension does not need to be typed.

CDB_File = Name of the Crash Database File. The file must have a *.txt* extension, but the extension does not need to be typed.

For example, to run Tippecanoe County (county number 79) for 1995 one would type

crash1 79 county79 env95

at the DOS prompt.

Screen Output

The following is a sample screen output from the program (line numbers for reference purposes only):

```
C:\temp>crash1 79 county79 env95                                (1)
Loading county79.txt into memory                                (2)
Total 729 subsegments in the Integration File                   (3)
Total 308 segments in the Integration File                      (4)

Processing data from env95.txt ..............................   (5)
..............................
Total 247531 crashes                                            (6)
Total 7783 crashes in county 79                                 (7)
Total number of missing psn      = 2126                         (8)
Total number of missing distance = 785                          (9)
Total number of missing direction = 9                           (10)

Total 4863 records to be checked for matched pair               (11)
No Pair    = 3063                                               (12)
Multi Pair = 331                                                (13)
One Pair   = 1469                                               (14)

Total 1469 records to be checked for Crash_Found                (15)
Misdir      = 19                                                (16)
Misdis      = 2                                                 (17)
Crashend    = 502                                               (18)
Crash_Found = 946                                               (19)
```

A description of the screen output is given in Table B.2.

Table B.2 Description of screen output

Line	Description
1	Command line to run program
2	Name of County File
3	Number of subsegments in County File
4	Number of segments (unique DRK's) in County File
5	Name of Crash Database File
6	Total number of crashes in Indiana in given year
7	Total number of crashes in county of interest
8	Number of crashes in county missed because of missing main street pseudo number, cross street pseudo number, or both
9	Number of crashes in county missed because of missing distance (given that both pseudo numbers are present)
10	Number of crashes in county missed because of missing direction (given that both pseudo numbers and distance are present)
11	Number of crash records in county containing main and cross street pseudo numbers, distance, and direction
12	No Pair indicates number of crashes in county with complete data but missed because the pseudo number pair could not be located in the County File (County File contains only Interstate, US, and State roads)
13	Multi Pair indicates number of crashes in county with complete data but missed because the pseudo number pair is not unique in the County File
14	One Pair indicates the number of crashes in the county with complete data and for which the pseudo number pair is unique in the County File
15	Same value as line 14
16	Number of crashes from line 15 missed because direction is coded incorrectly
17	Number of crashes from line 15 missed because distance is coded incorrectly
18	Number of crashes from line 15 which occurred near segment endpoints
19	Number of crashes from line 15 located on segments

Format of Intermediate Files

The program produces three intermediate files in text format: *temp1.txt*, *temp2.txt*, and *temp3.txt*. The file *temp1.txt* contains the crash records for those crashes which occur

in the county of interest, contain all data for crash location, and for which the pseudo number pair is unique in the County File. The file contains the same fields as the Crash Database File in the same order. For crashes occurring at intersections (*distref*=0), the *dirref* value has been changed from a '.' to a '9' to indicate direction not coded because the crash occurred at an intersection.

The file *temp2.txt* contains information for those crashes in *temp1.txt* after the location of the reference intersection and direction to search the County File have been determined. Each line corresponds to one crash. Each row contains the following information in sequential order: *Initial Position*, *Search Direction*, *severity*, *distref*, *road*, and DRK number corresponding to *Initial Position*. *Initial Position* indicates the County File record number containing the reference intersection given by the pseudo number pair in the Crash Database File record. (The first County File record has a Position value of 0). *Search Direction* gives the direction to search the County File from the reference intersection (1=move down in County File from reference intersection, -1=move up in County File from reference intersection, 0=unknown because direction was coded incorrectly).

The file *temp3.txt* contains the results for those crashes listed in *temp1.txt* and *temp2.txt*. Each line corresponds to one crash. The line contains the *Initial Position*, DRK number of the *Initial Position*, and the final result of the crash, which can be one of the following:

1) A message indicating that the direction was coded incorrectly (*Search Direction*=0).

2) A message indicating the *Final Position* and a DRK number corresponding to *Final Position*. This message indicates that the crash was assigned to a segment. *Final Position* indicates the County File record number after traversing *distref* from the reference intersection in the given *Search Direction*.

3) A message indicating the *Final Position* and indicating that the crash occurred near the endpoint of a DRK section. The crash is not assigned to a DRK segment because it occurred within approximately 30 m of the DRK segment endpoints.

The *temp.txt* files are overwritten each time the program is run. If the user would like to keep the *temp.txt* files, these files should be renamed or moved to another directory each time the program is run.

Format of Output Files

The program produces 2 output files in text format: *cXXCDB_File.o1* and *cXXCDB_File.o2*, where *XX* is the county number. For example, the output files for Tippecanoe County (county number 79) using the Crash Database File *env95.txt* would be *c79env95.o1* and *c79env95.o2*. The file *cXXCDB_File.o1* contains crash information for the segments. Each line corresponds to a different segment. The information in the columns are as follows (in order): DRK number for the segment, total number of crashes on the segment, number of fatal crashes on the segment, number of injury crashes on the segment, number of pdo crashes on the segment, and number of crashes on the segment for which severity was not coded.

The file *cXXCDB_File.o2* contains a list of summary statistics from the program run, including the number of crashes located and the number of crashes missed. These statistics include:

CDB_File	Name of Crash Database File.
Co_No	Number of county being run.
Int_File	Name of County File.
Crash_Tot	Total number of crashes in Indiana in given year.
Crash_in_Co	Total number of crashes in given county in given year.
Nopsn	Number of crashes missed because at least one pseudo number is missing in the crash record.
Nodist	Number of crashes missed because of missing distance, given that both pseudo numbers are present.
Nodir	Number of crashes missed because of missing direction, given that both pseudo numbers and distance are present.

No_Pair	Number of crashes with complete crash data but missed because pseudo number pair cannot be located in County File (County File only contains Interstate, US, and State roads).
Multi_Pair	Number of crashes with complete crash data but missing because pseudo number pair is not unique in County File (Pseudo number pair represents more than one location).
OnePairOnly	Number of crashes in given county with complete crash data and for which pseudo number pair is unique in County File.
Misdir	Number of crashes missed because direction is coded incorrectly (Search Direction in County File cannot be determined).
Misdis	Number of crashes missed because distance is coded incorrectly (Crash distance is greater than remaining distance on route in county).
Crashend	Number of crashes occurring near (within 30 m) the segment endpoints.
Crash_Found	Number of crashes located on DRK segments.

APPENDIX C: Negative Binomial Regression Output Using LIMDEP v 7.0

APPENDIX C: Negative Binomial Regression Output Using LIMDEP v 7.0

DEPENDENT VARIABLE: CRASH

```
--> negbin; lhs=crash
    ; rhs=one,lnlen,lnyrs,lnaadt,access,shldr,ps,twltl,nomedo
    ;rst=b0,1.0,1.0,1.0,b2,b3,b4,b5,b6,b7
    ;start=-1.1821,1.0,1.0,1.0,0.28495,-0.63087,2.5205,-0.74814,-
0.60432,1.1481 $

Normal exit from iterations. Exit status=0.

              +---------------------------------------------+
              | Negative Binomial Regression                |
              | Maximum Likelihood Estimates                |
              | Dependent variable              CRASH       |
              | Weighting variable                ONE       |
              | Number of observations            133       |
              | Iterations completed               12       |
              | Log likelihood function     -608.1492       |
              | Restricted log likelihood   -3145.620       |
              | Chi-squared                  5074.941       |
              | Degrees of freedom                  1       |
              | Significance level           .0000000       |
              +---------------------------------------------+
+---------+--------------+----------------+--------+---------+----------+
|Variable | Coefficient  | Standard Error |b/St.Er.|P[|Z|>z] | Mean of X|
+---------+--------------+----------------+--------+---------+----------+
 Constant -1.182147819      .29758445       -3.972   .0001
 LNLEN     1.000000000    ........(Fixed Parameter)........ .53812702
 LNYRS     1.000000000    ........(Fixed Parameter)........ 1.5825524
 LNAADT    1.000000000    ........(Fixed Parameter)........ 3.4307189
 ACCESS    .2849496209E-01  .84333366E-02    3.379   .0007  22.643631
 SHLDR    -.6308681891      .25697634       -2.455   .0141   .48120301
 PS        2.520481234      .91283083        2.761   .0058   .80370677E-01
 TWLTL    -.7481365395      .34333517       -2.179   .0293   .15037594
 NOMEDO   -.6043178204      .22811059       -2.649   .0081   .29323308
          Overdispersion parameter for negative binomial model
 Alpha     1.146810384      .13060917        8.780   .0000
```

DEPENDENT VARIABLE: PDO

```
--> negbin; lhs=pdo
    ; rhs=one,lnlen,lnyrs,lnaadt,access,shldr,ps,twltl,nomedo
    ;rst=b0,1.0,1.0,1.0,b2,b3,b4,b5,b6,b7
    ;start=-1.459,1.0,1.0,1.0,0.02613,-0.6691,2.627,-0.6859,-0.6843,1.105 $

Normal exit from iterations. Exit status=0.

              +---------------------------------------------+
              | Negative Binomial Regression                |
              | Maximum Likelihood Estimates                |
              | Dependent variable                 PDO      |
              | Weighting variable                 ONE      |
              | Number of observations             133      |
              | Iterations completed                11      |
              | Log likelihood function      -562.8458      |
              | Restricted log likelihood    -2404.700      |
              | Chi-squared                   3683.708      |
              | Degrees of freedom                   1      |
              | Significance level             .0000000     |
              +---------------------------------------------+
+---------+--------------+----------------+--------+---------+----------+
|Variable | Coefficient  | Standard Error |b/St.Er.|P[|Z|>z] | Mean of X|
+---------+--------------+----------------+--------+---------+----------+
 Constant -1.459314962      .27679502       -5.272   .0000
 LNLEN     1.000000000     ........(Fixed Parameter)........ .53812702
 LNYRS     1.000000000     ........(Fixed Parameter)........ 1.5825524
 LNAADT    1.000000000     ........(Fixed Parameter)........ 3.4307189
 ACCESS    .2613129338E-01  .81396886E-02    3.210   .0013   22.643631
 SHLDR    -.6690627205      .24964823       -2.680   .0074   .48120301
 PS        2.626807267      .86120851        3.050   .0023   .80370677E-01
 TWLTL    -.6858951756      .32147220       -2.134   .0329   .15037594
 NOMEDO   -.6843092643      .21162957       -3.234   .0012   .29323308
          Overdispersion parameter for negative binomial model
 Alpha     1.104513611      .13371908        8.260   .0000
```

DEPENDENT VARIABLE: FATINJ

```
--> negbin; lhs=fatinj
    ; rhs=one,lnlen,lnyrs,lnaadt,access,shldr,ps,twltl,nomedo
    ;rst=b0,1.0,1.0,1.0,b2,b3,b4,b5,b6,b7
    ;start=-4.923,1.0,1.0,1.0,0.0269,-0.4457,1.747,-0.4118,-0.5425,1.0 $

Normal exit from iterations. Exit status=0.
```

Negative Binomial Regression	
Maximum Likelihood Estimates	
Dependent variable	FATINJ
Weighting variable	ONE
Number of observations	133
Iterations completed	17
Log likelihood function	-454.2162
Restricted log likelihood	-1051.771
Chi-squared	1195.109
Degrees of freedom	1
Significance level	.0000000

Variable	Coefficient	Standard Error	b/St.Er.	P[|Z|>z]	Mean of X
Constant	-2.540095597	.31200507	-8.141	.0000	
LNLEN	1.000000000	........(Fixed Parameter)........			.53812702
LNYRS	1.000000000	........(Fixed Parameter)........			1.5825524
LNAADT	1.000000000	........(Fixed Parameter)........			3.4307189
ACCESS	.3245043915E-01	.77886695E-02	4.166	.0000	22.643631
SHLDR	-.5252034151	.25238133	-2.081	.0374	.48120301
PS	2.279538615	.86994394	2.620	.0088	.80370677E-01
TWLTL	-.8653717733	.35584344	-2.432	.0150	.15037594
NOMEDO	-.4934537038	.25230178	-1.956	.0505	.29323308
	Overdispersion parameter for negative binomial model				
Alpha	1.041156507	.13807634	7.540	.0000	

APPENDIX D: Code for Crash Extraction Software

APPENDIX D: Code for Crash Extraction Software

Computer program written by Nakarin Satthamnuwong.

```
/* CRASH1.CPP
  Objective : Extract crashes from Crash Database and
  place them on Road Inventory Segments.
  input : County NO.
        County Data File
        Environment Data file
  output: County_Environment.o1 and County_Environment.o2
  Tempfile: temp1.txt, temp2.txt, and temp3.txt
  Start       : Dec 22, 1997
  Last modify : Feb 21, 1998
  Nakarin Satthamnuwong, Purdue University
*/
#include <stdio.h>
#include <io.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
//Global Variable
double Enddis=1.9; //Distance near DRK segment endpoints within which a
crash
                                       //will not be counted on the
segment
int Crashend=0;    //number of crashes occurred near endpoints of DRK
section
int Crash_Found=0; //number of crashes located on the DRK section
int Crash_Tot=0;   //Total number of crashes in Crash Database;
int Crash_in_Co=0; //number of crashes in Co_No;
int Nopsn=0;       //missing at least one pseudo number (Main or Ref)
in CDB
int Nodist=0;      //missing distance in CDB
int Nodir=0;       //missing direction in CDB
int No_Pair=0;     //missed because psn pair can't be located in
Int_File
int Multi_Pair=0;  //missed because psn pair is not unique
int OnePairOnly=0; //unique pair in CDB_File and Int_File
int Misdis=0;      //missed because distance is coded incorrectly
int Misdir=0;      //missed because direction is coded incorrectly
struct IntRecord_type {
      long int DRK;      //DRK number of road section
      int Length;        //Length of section
      int Main_psn[4];   //Main Street pseudo numbers
      int Cross_psn[5];  //Cross Street pseudo numbers
      int Direction;     //Direction which sections are listed
};
struct DRKRecord_type {
      long int DRK;      //DRK number of road section
      int Total;       //number of crashes on the segment
      int Fatal;       //number of fatal crashes on the segment
      int Inj;         //number of injury crashes on the segment
      int Pdo;         //number of PDO crashes on the segment
```

```
      int Nosev;       //number of crashes with no severity on the
segment
};
int NumberDRK=0;      //numbers of different DRKs (MAX=2000)
struct DRKRecord_type DRKRec[2000];  //Record in DRK (MAX=2000)
int ScreenCDB(char CDB_Filename[],int Co_No,int NumberOfIntRec,struct
IntRecord_type *IntRec);
int FindNumberOfIntRec(char Int_Filename[]);
int LoadIntRec(char Int_Filename[],struct IntRecord_type *IntRec);
void InitDRKRec(int NumberOfIntRec,struct IntRecord_type *IntRec);
int CheckDirection(int NumberOfIntRec,struct IntRecord_type *IntRec);
int UpdateDRK(int NumberOfIntRec,struct IntRecord_type *IntRec);
int SaveOutputFile(char Filename[],char F1[],int Co_No,char F2[]);
int LocateSiglePair(int NumberOfIntRec,struct IntRecord_type
*IntRec,int Main,int Cross);
int convert(int Direction);
int Similar(int Dira,int Dirb);

int main(int argc, char * argv[]) {
       if (argc != 4) {
            printf("How to use the program\n");
            printf("type: crash1 County_NO county_filename
environment_filename\n");
            printf("      All data files must have extension .TXT\n");
            printf("example : crash1 79 county79 env95\n");
            printf("The output will be saved in 2 files: c79env95.o1
and c79env95.o2\n");
            return(-2);
      }
      char CDB_Filename[80];        //Crash Database file
      char Int_Filename[80];        //Integration File
      int Co_No;                    //County Number
      int NumberOfIntRec=0;                  //Total number record in
Int_File
      struct IntRecord_type *IntRec;         //Record in Int_File
      struct DRKRecord_type *DRKRec;         //Record in DRK
      Co_No=atoi(argv[1]);
      strcpy(CDB_Filename,argv[3]);strcat(CDB_Filename,".txt");
      strcpy(Int_Filename,argv[2]);strcat(Int_Filename,".txt");
      //reset temp1.txt temp2.txt temp3.txt
      printf("Loading %s into memory\n",Int_Filename);
      NumberOfIntRec = FindNumberOfIntRec(Int_Filename);    //find
NumberOfIntRec
      if (NumberOfIntRec==-1) return (-1);                  //FILE NOT
FOUND
      printf("Total %d subsegments in the Integration
File\n",NumberOfIntRec);
      IntRec = new IntRecord_type [NumberOfIntRec];//allocate memory
for Int_File
      if (IntRec==NULL) {printf("\nNot enough memory");return(-1);}
      LoadIntRec(Int_Filename,IntRec); //Load Int_file into memory
      InitDRKRec(NumberOfIntRec,IntRec);
      printf("Total %d segments in the Integration
File\n\n",NumberDRK);
      printf("Processing data from %s ",CDB_Filename);
```

```
     if (ScreenCDB(CDB_Filename,Co_No,NumberOfIntRec,IntRec)==-1)
return(-1);
     printf("\n");
     printf("Total %d crashes\n",Crash_Tot);
     printf("Total %d crashes in county%3d\n",Crash_in_Co,Co_No);
     printf("Total number of missing psn       = %d\n",Nopsn);
     printf("Total number of missing distance  = %d\n",Nodist);
     printf("Total number of missing direction = %d\n\n",Nodir);
     printf("Total %d records to be checked for matched
pair\n",Crash_in_Co-Nopsn-Nodir-Nodist);
     printf("No Pair     = %d\n",No_Pair);
     printf("Multi Pair  = %d\n",Multi_Pair);
     printf("One Pair    = %d\n\n",OnePairOnly);
     printf("Total %d records to be checked for
Crash_Found\n",OnePairOnly);
     CheckDirection(NumberOfIntRec,IntRec); //save temp2.txt
     printf("Misdir      = %d\n",Misdir);
     UpdateDRK(NumberOfIntRec,IntRec);
     printf("Misdis      = %d\n",Misdis);
     printf("Crashend    = %d\n",Crashend);
     printf("Crash_Found = %d\n",Crash_Found);
   char OutputfileName[80]="c";
   strcat(OutputfileName,argv[1]);strcat(OutputfileName,argv[3]);
     SaveOutputFile(OutputfileName,CDB_Filename,Co_No,Int_Filename);
  return 0;
} //end Main

int ScreenCDB(char CDB_Filename[],int Co_No,int NumberOfIntRec,struct
IntRecord_type *IntRec) {
     FILE *CDB_File,*Temp_File;
     //OPEN DATA FILES//
     if ((CDB_File = fopen(CDB_Filename,"r")) == NULL)
      {printf("\nCan't find %s.\n",CDB_Filename);return(-1);}
     if ((Temp_File = fopen("temp1.txt","w")) == NULL)
      {printf("\nCan't write temp1.txt\n");return(-1);}
     //Start Reading CDB_File & save only records with county = Co_No,
     //complete crash data, and unique pair to temp1.txt
     do {
           char
str1[100]="",t1[50],t2[50],t3[50],t4[50],t5[50],t6[50];
           fgets(str1,90,CDB_File);
           if (strlen(str1)==0) break;
           Crash_Tot++;
           sscanf(str1,"%s %s %s %s %s %s",t1,t2,t3,t4,t5,t6);
           //t1=county of Accident,t2= Severity,t3=Main_Street_PSN,
           //t4=Cross_Street_PSN, t5=CDB_Dir, t6=CDB_Dist
           if (atoi(t1)==Co_No) {
     Crash_in_Co++;
     if (t3[0]=='0' || t4[0]=='0') Nopsn++;
           else {
             if (t6[0]=='0' ) t5[0]='9';
             if (t6[0]=='.' ) Nodist++;
           else if (t5[0]=='.' ) Nodir++;
             else {// check matched Pair
                  int PAIR=0;
```

```
                    for (int i=0;i<NumberOfIntRec;i++) {
                     int Test1=0,Test2=0,j;
                     //Check Main_Street_PSN
                     for (j=0;j<4;j++) if
(atoi(t3)==IntRec[i].Main_psn[j]) Test1=1;
                     if (Test1==1) //Check Cross_Street_PSN if
Main_Street_PSN matches
                          for (j=0;j<5;j++) if
(atoi(t4)==IntRec[i].Cross_psn[j]) Test2=1;
                     if (Test2==1) PAIR++; //Both PSNs match
                    }//end for i
                    if (PAIR==0) No_Pair++;
                    if (PAIR>=2) Multi_Pair++;
                    if (PAIR==1) {   //Record in county with complete
location data and
                                                      //unique pair in
Integration File
                          fprintf(Temp_File,"%s %s %s %s %s
%s\n",t1,t2,t3,t4,t5,t6);
                          OnePairOnly++;
                    };
               } //end else
         } //end else if (t3[0]=='0' || t4[0]=='0') Nopsn++;
       } //end if in Co_NO
       div_t x=div(Crash_Tot,2500);if (x.rem==0) printf(".");
   } while (!feof(CDB_File));
   //Close data file
      fclose(Temp_File);fclose(CDB_File);
   return 0;
} // end function ScreenCDB

int FindNumberOfIntRec(char Int_Filename[]) {
      int NumberOfIntRec=0;
   FILE *Int_File;
   if ((Int_File = fopen(Int_Filename,"r")) == NULL)
       {printf("\nCan't find %s.\n",Int_Filename);return(-1);}
   //Find number of records in Integration File
      do {
      char str1[200]="";
      fgets(str1,190,Int_File);
            if (strlen(str1)==0) break;
      if (strlen(str1)>30) NumberOfIntRec++;
   } while (!feof(Int_File));
   fclose(Int_File);
      return NumberOfIntRec;
}//end of FindNumberOfIntRec

int LoadIntRec(char Int_Filename[],struct IntRecord_type *IntRec) {
   FILE *Int_File;
   if ((Int_File = fopen(Int_Filename,"r")) == NULL)
       {printf("\nCan't find %s.\n",Int_Filename);return(-1);}
   //loading Int datafile into memory
      int Count = 0;
   do {
      char str1[200]="";
```

```
          fgets(str1,199,Int_File);
      if (strlen(str1)==0) break;
      if (strlen(str1)>30) {
      long int t[20];
      sscanf(str1,"%d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d
%d",&t[1],&t[2],&t[3],&t[4],&t[5],&t[6],&t[7],&t[8],&t[9],&t[10],&t[11]
,&t[12],&t[13],&t[14],&t[15],&t[16],&t[17]);
      IntRec[Count].DRK=t[1];
      IntRec[Count].Length=t[6];
      IntRec[Count].Main_psn[0]=t[8];
      IntRec[Count].Main_psn[1]=t[9];
      IntRec[Count].Main_psn[2]=t[10];
      IntRec[Count].Main_psn[3]=t[11];
      IntRec[Count].Cross_psn[0]=t[12];
      IntRec[Count].Cross_psn[1]=t[13];
      IntRec[Count].Cross_psn[2]=t[14];
      IntRec[Count].Cross_psn[3]=t[15];
      IntRec[Count].Cross_psn[4]=t[16];
      IntRec[Count].Direction   =t[17];
      Count++;
     }
   } while (!feof(Int_File));
   fclose(Int_File);
   return 0;
} //end of LoadIntRec

int CheckDirection(int NumberOfIntRec,struct IntRecord_type *IntRec) {
      FILE *Temp_File1,*Temp_File2;
   //OPEN DATA FILES//
   if ((Temp_File1 = fopen("temp1.txt","r")) == NULL)
    {printf("\nCan't read temp1.txt\n");return(-1);}
   if ((Temp_File2 = fopen("temp2.txt","w")) == NULL)
    {printf("\nCan't write temp2.txt\n");return(-1);}
   //Read data in temp1.txt
      do {
      char str1[100]="";
      int t1=0,t2=0,t3=0,t4=0,t5=0,t6=0,CheckPrev=0,j;
            fgets(str1,90,Temp_File1);
      if (strlen(str1)==0) break;
            sscanf(str1,"%d %d %d %d %d %d",&t1,&t2,&t3,&t4,&t5,&t6);
      //t1=county of Accident,t2= Severity
            //t3=Main_Street_PSN,t4=Cross_Street_PSN
            //t5=CDB_Dir, t6=CDB_Dist      <<<<<<<<<<< TEMP1.TXT FORMAT
      int Position,SearchDirection=0;
      Position=LocateSiglePair(NumberOfIntRec,IntRec,t3,t4);
      if (t5==9) SearchDirection=1; //crash occurred at intersection
      else {
      //Check to see if Int_File record Position-1 contain CDB_Main_PSN
      for (j=0;j<4;j++) if (t3==IntRec[Position-1].Main_psn[j])
CheckPrev=1;
      if (CheckPrev==0) {
        if (Similar(IntRec[Position].Direction,t5)) SearchDirection=1;
      } else {
        int UpDir,Test1,Test2;
        if (IntRec[Position-1].Direction < 4)
```

```
                 UpDir=IntRec[Position-1].Direction+4;
            else UpDir=IntRec[Position-1].Direction-4;
        Test1=Similar(IntRec[Position].Direction,t5);
        Test2=Similar(UpDir,t5);
        if (Test1==1 && Test2==0) SearchDirection=1;
        if (Test1==0 && Test2==1) SearchDirection=-1;
      } //end  if (CheckPrev==0)
      } //end if (t5==9)
            if (SearchDirection==0) Misdir++;
      fprintf(Temp_File2,"%5d %2d %2d %5d %9d %12d\n",

Position,SearchDirection,t2,t6,t3,IntRec[Position].DRK);
   } while (!feof(Temp_File1));
      //CLOSE DATA FILES//
   fclose(Temp_File1);fclose(Temp_File2);
      return 0;
} //end CheckDirection

int LocateSiglePair(int NumberOfIntRec,struct IntRecord_type
*IntRec,int Main,int Cross) {
      int i,j;
      for (i=0;i<NumberOfIntRec;i++) {
            int Test1=0,Test2=0;
            for (j=0;j<4;j++) if (IntRec[i].Main_psn[j]==Main) Test1=1;
            if (Test1==1)
                  for (j=0;j<5;j++) if (IntRec[i].Cross_psn[j]==Cross)
Test2=1;
            if (Test2==1) break;    //Single matching pair found
  } //end for i
  return i;
} //end LocateSiglePair

int convert(int Direction) {
  int NewDir=Direction;
  switch (Direction) {
      case 1:NewDir=2;break;case 2:NewDir=1;break;
      case 5:NewDir=6;break;case 6:NewDir=5;break;
  }
  return NewDir;
}//end convert

int Similar(int Dira,int Dirb) {
  int Dir1,Dir2,t1,t2;
  Dir1=convert(Dira);Dir2=convert(Dirb);
  t1=Dir1+1;t2=Dir1-1;
  if (t1== 8) t1=0;
  if (t2==-1) t2=7;
  if (Dir2==Dir1 || Dir2==t1 || Dir2==t2) return 1; else return 0;
} //end Similar

void InitDRKRec(int NumberOfIntRec,struct IntRecord_type *IntRec) {
  int i,j;
  for (i=0;i<2000;i++) {
   DRKRec[i].DRK=0;DRKRec[i].Total=0;DRKRec[i].Fatal=0;
   DRKRec[i].Inj=0;DRKRec[i].Pdo=0;DRKRec[i].Nosev=0;
```

```
  } //end for i
  for (i=0;i<NumberOfIntRec;i++) {
      int PASS=0;
            for (j=0;j<NumberDRK;j++) if (DRKRec[j].DRK==IntRec[i].DRK)
PASS=1;
      if (PASS==0) {
      DRKRec[NumberDRK].DRK=IntRec[i].DRK;
      NumberDRK++;
      } //end if
  }//end for i
}//end InitDRKRec

int UpdateDRK(int NumberOfIntRec,struct IntRecord_type *IntRec) {
      FILE *Temp_File3,*Temp_File2;
   //OPEN DATA FILES//
   if ((Temp_File2 = fopen("temp2.txt","r")) == NULL)
    {printf("\nCan't read temp2.txt\n");return(-1);}
   //Read data in temp2.txt
   if ((Temp_File3 = fopen("temp3.txt","w")) == NULL)
    {printf("\nCan't write temp3.txt\n");return(-1);}
   //Read data in temp2.txt
   for (int Loop=0;Loop<OnePairOnly;Loop++) {
            char str1[100]="";
      int Main_Street_PSN,Position,SearchDir,Severity,CDB_Dist,PASS1;
      double Crash_Dist,Totlen=0;
            fgets(str1,90,Temp_File2);
      if (strlen(str1)==0) break;
      sscanf(str1,"%d %d %d %d %d",
      &Position,&SearchDir,&Severity,&CDB_Dist,&Main_Street_PSN);
            fprintf(Temp_File3,"Begin at %4d (DRK %d)
",Position,IntRec[Position].DRK);
      if (SearchDir==0) {
        fprintf(Temp_File3,"** Direction incorrect (Misdir) **\n");
        continue;
      } //end if (SearchDir==0)
      //round the crash_Dist to the nearest 0.1
      Crash_Dist=CDB_Dist/52.8;
      if (Crash_Dist*10-floor(Crash_Dist*10) < 0.5)
      Crash_Dist=floor(10*Crash_Dist)/10.0; else
Crash_Dist=ceil(10*Crash_Dist)/10.0;
      //correct the search position (for up direction)
      if (SearchDir==-1) Position=Position-1;
            do { //add segment lengths to find crash location
       Totlen=Totlen+IntRec[Position].Length;
       PASS1=0; // start PASS1=0
       for (int i=0;i<4;i++)  //Verify that length added is on
                                                        //route where
crash occurred
            if (IntRec[Position].Main_psn[i]==Main_Street_PSN) PASS1=1;
       Position=Position+SearchDir;
       if (PASS1==0) {
            fprintf(Temp_File3,"End at %4d **Check Main_Street_PSN fail
(Misdis)**\n",Position-SearchDir);
            break;}
            } while (Crash_Dist > Totlen);
```

```
            if (PASS1==0) {Misdis++;continue;} // go to the next record
      Position=Position-SearchDir;
       //check the Crashend condition to see if crash occurred near
       //DRK segment endpoints
      double tmp1=Crash_Dist-Totlen+IntRec[Position].Length;
      double tmp2=Totlen-Crash_Dist;
            //tmp1 and tmp2 are distances to endpoints of subsegment
            if (fabs(tmp1-Enddis) < 1e-10) tmp1=Enddis;
      if (fabs(tmp2-Enddis) < 1e-10) tmp2=Enddis;
      if ((tmp1<=Enddis && IntRec[Position].DRK !=IntRec[Position-
SearchDir].DRK) ||
        (tmp2<=Enddis &&
IntRec[Position].DRK!=IntRec[Position+SearchDir].DRK))
            {Crashend++;fprintf(Temp_File3,"End at %4d **near end point
(Crashend)**\n",Position);
       continue;} // go to the next record
     // Crash_Found >>>> update DRK
      Crash_Found++;
      for (int j=0;j<NumberDRK;j++) {
         if (DRKRec[j].DRK==IntRec[Position].DRK) {
            DRKRec[j].Total++;
            switch (Severity) {
             case 0:DRKRec[j].Nosev++;break;
             case 1:DRKRec[j].Fatal++;break;
             case 2:DRKRec[j].Inj++;break;
             case 3:DRKRec[j].Pdo++;break;
            } //end switch
            break;
         };//end if (DRKRec[j].DRK==IntRec[Position].DRK)
      } //end (int j=0;j<NumberDRK;j++)
      fprintf(Temp_File3,"End at %4d (DRK
%d)\n",Position,IntRec[Position].DRK);
   } // end for LOOP;
   //CLOSE DATA FILES//
   fclose(Temp_File2);fclose(Temp_File3);
      return 0;
} //end UpdateDRK

int SaveOutputFile(char Filename[],char F1[],int Co_No,char F2[]) {
   FILE *OUT1,*OUT2;
      char Filename1[80],Filename2[80];
      strcpy(Filename1,Filename);strcat(Filename1,".01");
   strcpy(Filename2,Filename);strcat(Filename2,".02");
   //OPEN DATA FILES//
   if ((OUT1 = fopen(Filename1,"w")) == NULL)
    {printf("\nCan't write %s\n",Filename1);return(-1);}
   if ((OUT2 = fopen(Filename2,"w")) == NULL)
    {printf("\nCan't write %s\n",Filename2);return(-1);}
   //WRITE OUTPUT FILE 1
   for (int i=0;i<NumberDRK;i++)
      fprintf(OUT1,"%10d %4d %4d %4d %4d
%4d\n",DRKRec[i].DRK,DRKRec[i].Total,

      DRKRec[i].Fatal,DRKRec[i].Inj,DRKRec[i].Pdo,DRKRec[i].Nosev);
   //WRITE OUTPUT FILE 2
```

```
   fprintf(OUT2,"%s\n%d\n%s\n",F1,Co_No,F2);
   fprintf(OUT2,"%d\n%d\n",Crash_Tot,Crash_in_Co);
      fprintf(OUT2,"%d\n%d\n%d\n%d\n%d\n",Nopsn,Nodist,Nodir,No_Pair,Mu
lti_Pair);
      fprintf(OUT2,"%d\n%d\n%d\n%d\n%d\n",OnePairOnly,Misdir,Misdis,Cra
shend,Crash_Found);
   fclose(OUT1);fclose(OUT2);
   return 0;
} //end SaveOutputFile
```